
Table of Contents

Images	110
Code samples	115
Labels	116
Links	117
Navtabs	120
Tables	124
Syntax highlighting	128
Workflow maps	131

Handling reviews

Commenting on files	135
---------------------------	-----

Publishing

Build arguments	138
Themes	141
Generating PDFs	142
Help APIs and UI tooltips	154
Search configuration	166
iTerm profiles	170
Pushing builds to server	172


```
@SLBJC CVCA HCI WJJ QCPTC
```

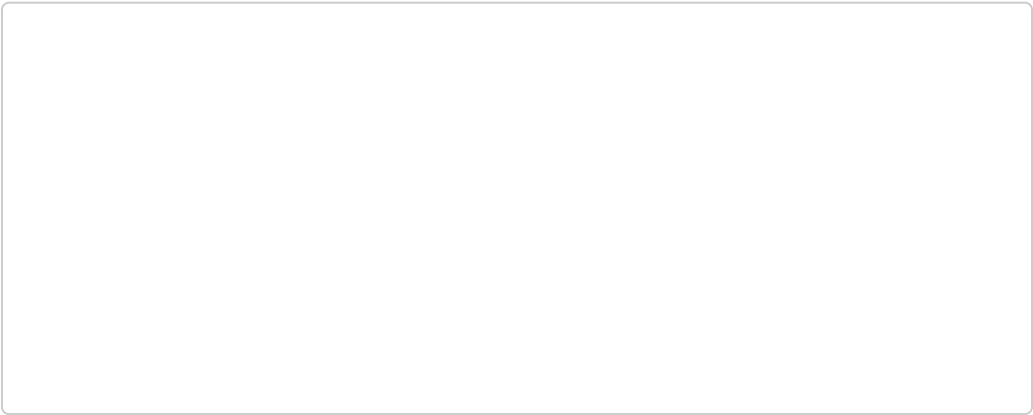
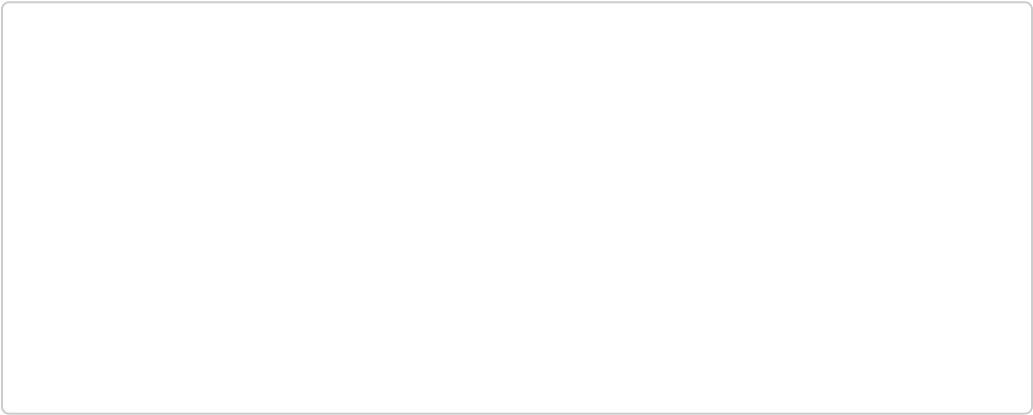
If you want to shorten this long command, you can put this code in a file such as `jekyll.sh` (on a Mac) and then simply type `HCI WJJ QF` to build Jekyll.

Running the site in Docker

You can also use Docker to directly build and run the site on your local machine. Just `4569a534repold` and run `thfollownninfromon youworknnino d:e`.

```
@YWB LM G?Geø MG OKNM Q @^C
```

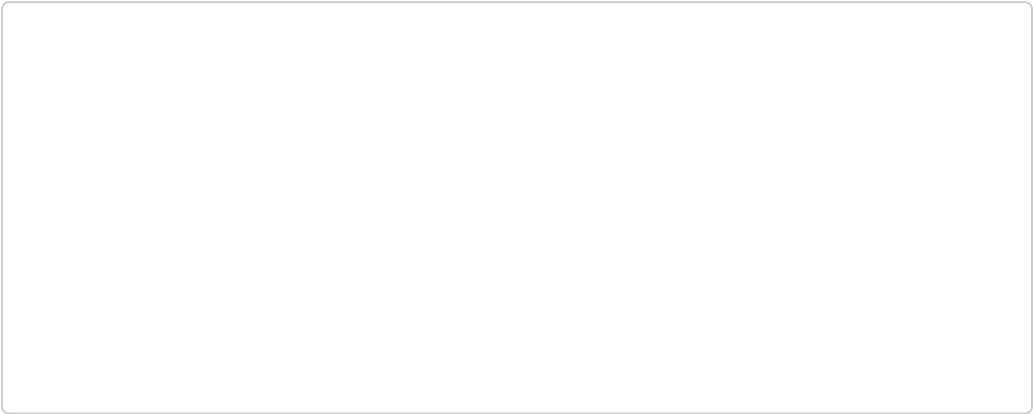
The `QGBC@?P KWBMA=QGBC@?P` refers8.07 the `_data/sidebars/mydoc_sidebar.yml`



Top navigation

The top navigation works just like the sidebar. You can specify which topnav data file should load by adding a `RMNL?T` property in your page, like this:

RMNL?T



CLRPGCQ

RGRJC QGBC@?P

NPMBSAR *CIWJJ \$MA 4FCKC

TCPOGML

DMJBCPQ

RGRJC /TCPTGCU

MSBBSR NBD@ NBD

DMJBCPGRCKQ

RGRJC ' CR QR?PRCB

SPGLBCGLKJ 9KJ

CLRPGCQ

RGRJC QGBC@?P

NPMB SAR *CI WJJ \$MA 4FCKC

TCPOGML

DMJB CPQ

RGRJC /TCPTGCU

MSRNSR UC@ NBD

DMJB CPGRCKQ

RGRJC 2CJC?QC .MRCQ

MSRNSR UC@ NBD

DMJB CPGRCKQ

RGRJC 2CJC?QC LMRCQ

SPJ KWBMA=PCJC?QC=LMRCQ= FRKJ

MSRNSR UC@ NBD

RGRJC 2CJC?QC LMRCQ

SPJ KWBMA=PCJC?QC=LMRCQ= FRKJ

MSRNSR UC@ NBD

RGRJC)LQR?JJ?RGML

MSRNSR UC@ NBD

DMJB CPGRCKQ

RGRJC !@MSR 2S@W 'CKQ "SLBJCP CRA

SPJ KWBMA=?@MSR=PS@W=ECKQ=CRA FRKJ

MSRNSR UC@ NBD

RGRJC)LQR2@9W?Q1WJJ-MA -?A

SPJ QKWBMA@P!@00480\$00480QBMA@MSRNSR UC@ %GCU

```
RGRJC
MSRNSR  NBD
RWNC  DPMLRK?RRCP
DMJBCPGRCKQ
  RGRJC
  SPJ  RGRJCN?EC
  MSRNSR  NBD
  RWNC  DPMLRK?RRCP
  RGRJC
  SPJ  RMAN?EC
  MSRNSR  NBD
  RWNC  DPMLRK?RRCP
```

Leave the output as `MSRNSR NBD` for these frontmatter pages so that they don't appear in the web output.

For more detail on the sidebar, see [Sidebar navigation \(page 71\)](#) and [YAML tutorial \(page 74\)](#).

Relative links and offline viewing

This theme uses relative links throughout so that you can view the site offline and not worry about which server or directory you're hosting it. It's common with tech docs to push content to an internal server for review prior to push4667tu4

RGRJC 3MKC RGRJC
R?EQ 9Q?KNJC Q?KNJC ;
I CWUMPBQ I CWUMPB I CWUMPB I CWUMPB
J?QR=SNB?RCB -MLRF B?W WC?P
QSKK?PW MNRGML?J QSKK?PW

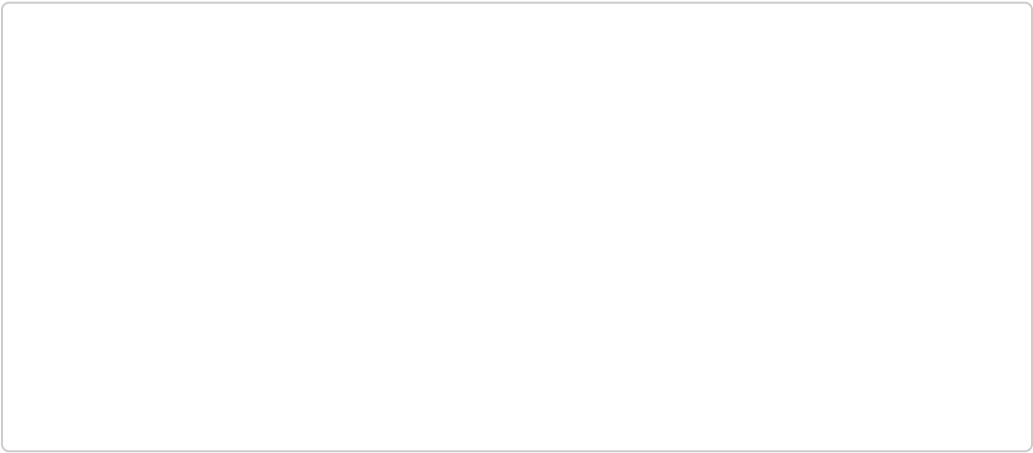
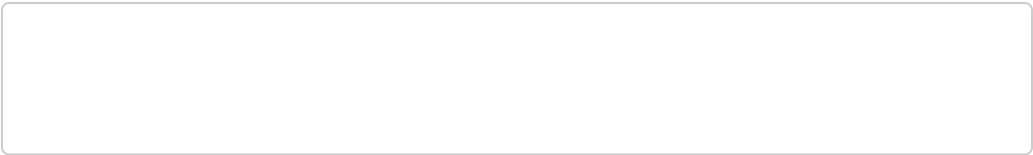
Configure the top navigation

The top navigation bar's menu items are set through the `_data/topnav.yml` file. Use the top navigation bar to provide links for navigating from one product to another, or to navigate to external resources.

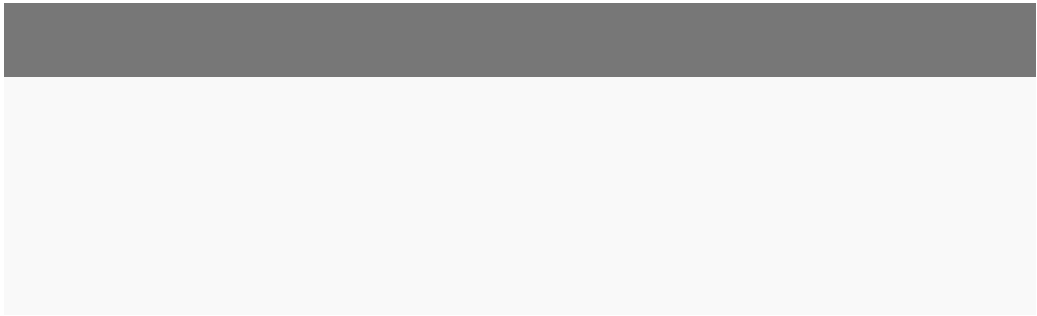
For external URLs, use



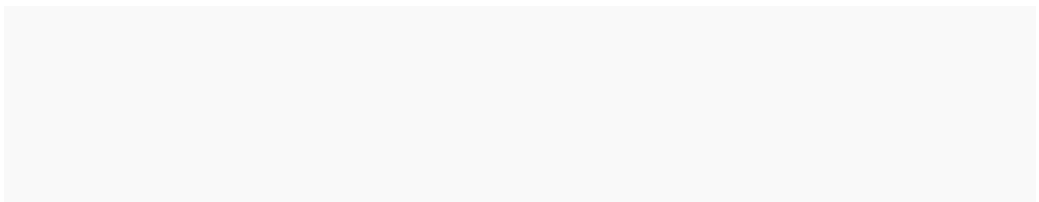
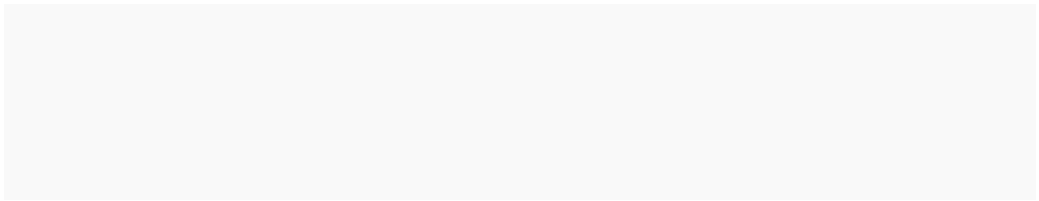
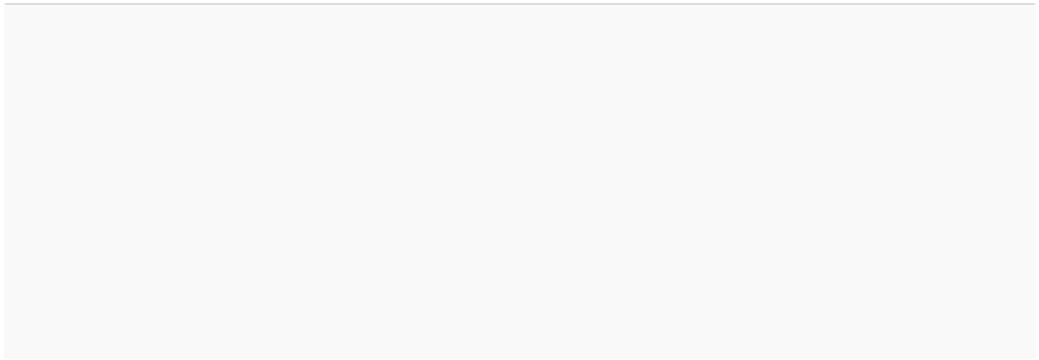
The news/news.html file displays the postso0024BT0.2000oe postso00_archivedisplays thshows



Other instructions



Features	Supported	Notes
Blogging features	Yes	



About the theme's author

Summary: I have used this theme for projects that I've worked on as a professional technical writer.

My name is Tom Johnson, and I'm arofessional tked rloggked , anpodcastkeas

[myloggk's about page](#)

. See

[course on API documentation](#)

figh, you cop ewek, trying to get like, be the place to improve ing you with the theme 'w

balling, you cop ewek, trying to get like, be the place to improve ing you with the theme 'w

Support

| Summary: Contact me for any support issues.

Let me know about any bugs or other issues that you find. Just email me at tomjoht@gmail.com. You can also [create issues directly within the Github repository here](#).

Release notes 6.0



Summary: Version 6.0 of the Documentation theme for Jekyll,

' %-

PCKMRC FRRNQ PS@WECKQ MPE

QNCAQ

2CB#JMRF

ŠWDSRÁÖŦ Å5öÅ \$
ŽWJÅ5ôä ' @

M4u8JG9\

KCPACL?PW
KGLG=NMPRGJC
KGLGRCQR
KSJRGN?PR NMQR
LCR BLQ
LMI MEGPG
KGLG=NMPRGJC \ PA
MARM GR
Q?UWCP \
NS@JGA=QSDDGV
P@ DQCTCLR
P@ GLMRGDW
DDG
PBGQAMSLR
PCBA?PNCR
PMSEC
Q?DC=W?KJ
Q?QQ
0AUWOP9D\$ q\$5 000! 5' opQ? SJC \

Summary: Installation of Jekyll on Mac is usually less problematic than on Windows. However, you may run into permissions issues with Ruby that you must overcome. You should also use Bundler to be sure that you have all the required gems and other utilities on your

Ruby and RubyGems

Ruby and [RubyGems](#) are usually installed by default on Macs. Open your Terminal and type

[Bundler \(page 37\)](#) section.

However, if your location is something like `5QCPQ -?A"MMI OPM PTK PS@GCQ`

Install Homebrew

Homebrew is a package manager for the Mac, and you can use it to install an alternative instance of Ruby code. To install Homebrew, run this command:

```

  SQP @GL PS@W  C      ASPJ  DQ3, FRRNQ  P?U EGRFS@SQCPAMLRCLR AM
K (MKC@PCU GLQR?JJ K?QRCP GLQR?JJ

```

If you already had Homebrew installed on(If11.8pulte, be suret toupdats id:) TjETBT120.37504940.6723 T

SQP M ?@ @GL P\$@S
rewriteable.n

3. In your `.bash_profile`

Install Jekyll on Windows

Tip:

7. Type `PS@W BI P@ GLQR?JJ`

If you get stuck, see the [official instructions for installing Ruby Dev Kit](#) .

Installeae5xekyallgemt

Now uspe

```
QMSPAC  FRRNQ  PS@WECKQ MPE
```

```
ECK  UBK
```

```
ECK  HCI WJJ
```

The [wdm gem](#) allows for the polling of the directory and rebuilding of the Jekyll site when you make changes. This gem is needed for Windows users, not Mac users.

5. Save and close the file.

6. Type `bundle install`.

Bundle retrieves all the needed gems and gem dependencies and downloads them to your computer. At this time, Bundle also takes a snapshot of all the gems used in your project and creates a Gemfile.lock file to store this information.

Git Clients for Windows

Although you can use the default command prompt with Windows, it's recommended that you use [Git Bash](#) instead. The Git Bash client will allow you to

If so, you will need to take some additional steps to resolve it. (Note that this error only appears if you have the `github-pages` gem in your `gemfile`.) The resolution involves adding a Github token and a cert file.

See this post on [Knight Codes](#) for instructions on how to fix the error. You basically generate a personal token on Github and set it as a system variable. You also download a certification file and set it as a system variable. This resolves the issue.

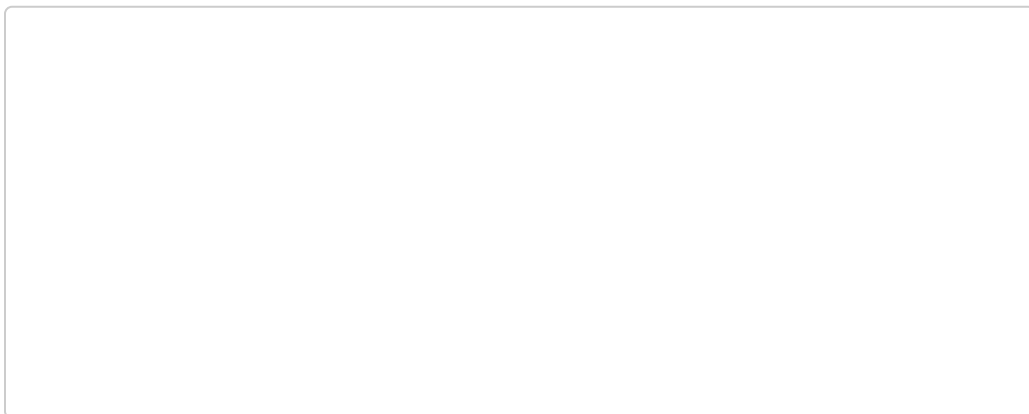
Pages

Summary: This theme primarily uses pages. You need to make sure your pages have the appropriate frontmatter. One frontmatter tag your users might find helpful is the summary tag. This functions similar in purpose to the shortdesc element in DITA.

Where to author content

Use a text editor such as Sublime Text, WebStorm, IntelliJ, or Atom to create pages. Atom is recommended because it's created by Github, which is driving some of the Jekyll development through Github Pages.

Where to save pages



Frontmatter is always formatted with three hyphens at the top and bottom. Your frontmatter must have a `RGRJC` and `NCPK?JGLI` value. All the other values are optional.

Note that you cannot use variables in frontmatter.

The following table describes each of the frontmatter that you can use with this theme:

Frontmatter	Required?	Description
title	Required	The title for the page
tags	Optional	Tags for the page. Make all tags single words,

Frontmatter	Required?	Description
summary	Optional	A 1-2 word sentence summarizing the content on the page. This gets added into the (summary) TJE

If you have a lot of HTML, as long as the top and bottom tags of the HTML are flush left in a Markdown file, all the tags inside those bookend HTML tags will render as HTML, regardless of their indentation. (This can be especially useful for tables.)

Page names

I recommend prefixing your page names with the product, such as “mydoc_pages” instead of just “pages.” This way if you have other products that also have topics with generic names such as “pages,” there won’t be naming conflicts.

Additionally, consider adding the product name in parentheses after the title, such as “Pages (Mydoc)” so that users can clearly navigate different topics for each product.

Kramdown Markdown

Kramdown is the Markdown flavor used in the theme. This mostly aligns with Github-flavored Markdown, but with some differences in the indentation allowed within lists. Basically, Kramdown requires you to line up the indent between list items with the first starting character after the space in your list item numbering. See this [blog post on Kramdown and Rouge](#) for more details.

You can use standard Multimarkdown syntax for tables. You can also use fenced code blocks with lexers specifying the type of code. The configuration file shows the Markdown processor and extension:



(C?BGLEQ UGRF)\$ 4?EQ Y QMKC)B4?E[

I

Frontmatter	Required?	Description
tags	Optional	Tags for the page. Make all tags single words, with underscores if needed. Separate them with commas. Enclose the whole list within brackets. Also, note that tags must be added to <code>_data/tags_doc.yml</code> to be allowed entrance into the page. This prevents tags from becoming somewhat random and unstructured. You must create a tag page for each one of your tags following the sample pattern in the <code>tags</code> folder. (Tag pages aren't automatically created.)
keywords	Optional	Synonyms and other keywords for the page. This information gets stuffed into the page's metadata to increase SEO. The user won't see the keywords, but if you search for one of the keywords, it will be picked up by the search engine.
sidebar	Required	Refers to the sidebar data file for this page. Don't include the <code>.yml</code> file extension for the sidebar — just provide the file name. If no sidebar is specified, this value will inherit the <code>BCD?SJR</code> property

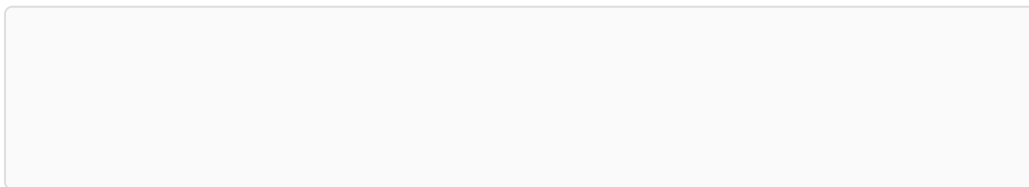
Complex Lists

Here's a more complex list:

3?KNJC DGPQR GRCK

Key Principle to Remember with Lists

Conditional logic



Or operator

You can use more advanced Liquid markup for conditional logic, such as an `MP` command. See [Shopify's Liquid documentation](#) for more details.

31.example, here's an.example using


```
RFGLE    YYN?EC RFGLE  [[  
RFGLE    YYN?EC RFGLE  [[
```

I use includes all the time. Most of the includes in the _includes directory are

Video tutorial on collections

See this [video tutorial on Jekyll.tips](#) for more details on collections.

WebStorm Text Editor

Summary: You can use a variety of text editors when working with a Jekyll project. WebStorm from IntelliJ offers a lot of project-specific features, such as find and replace, that make it ideal for working with tech comm projects.

About text editors and WebStorm

There are a variety of text editors available, but I like WebStorm the best because it groups files into projects, which makes it easy to find all instances of a text string, to do find and replace operations across the project, and more.

If you decide to use WebStorm, here are a few tips on configuring the editor.

Remove unnecessary plugins

By default, WebStorm comes packaged with a lot more functionality than you probably need. You can lighten the editor by removing unnecessary plugins.

Enable Soft Wraps (word wrapping)

Most likely you'll want to enable soft wraps, which wraps lines rather than extending them out forever and requiring you to scroll horizontally to see the text. To enable softwrapping, go to WebStorm > Preferences > Editor > General and see the Soft Wraps section. Select the Use soft wraps in editor check box.

Exclude a directory

When you're searching for content, you don't want to edit any file that appears in the `_site` directory. You can exclude a directory from Webstorm by right-clicking

[illegible]

Atom Text Editor



1

There's a danger with setting the accordion to

YAML tutorial in the context of Jekyll



Summary: YAML is a format that relies on white spacing to separate

Feedback

This is my feedback to you. Even if I include linebreaks here, all of the linebreaks will be removed when the value is inserted.

- road bikes
- hybrid bikes

Here we use a “for” loop to get each item in the bikes list. By using `RGRJC` we only get the `RGRJC` property from each list item.

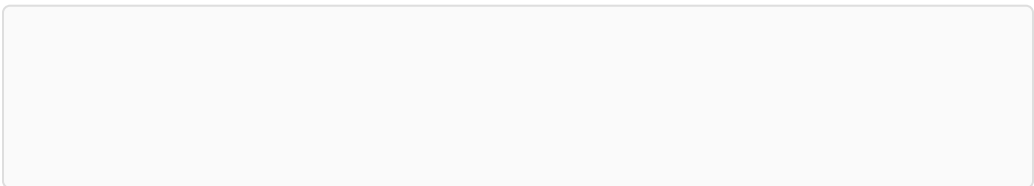
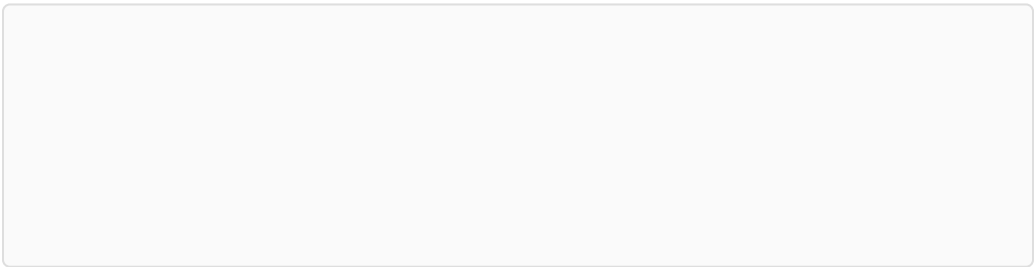
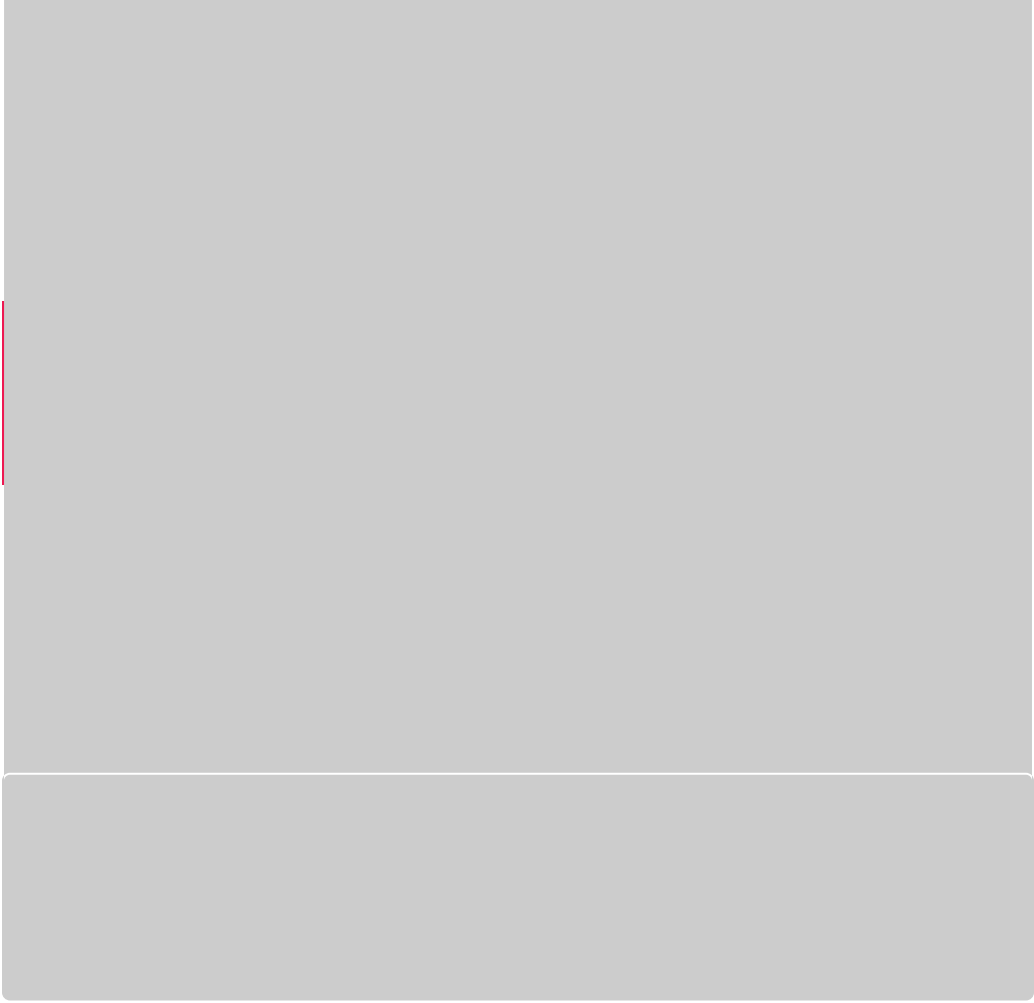
YAML:

```
Q?JCQRC?KQ
```



Thh2 gxample 2 gnotably different. Here I'm showing how toreuse content in YAML

KWN?ECQ



Fo30 gimplicity, make all your tags gingle words (connect them with hyphens if

3. Change the title, tagName, and permalink values to be specific to the tag name you just created.

By default, the `_layouts/page.html` file will look for any tags on a page and insert them at the bottom of the page using this code:

```
BGT AJ?QQ R?EQ
Y if N?EC R?EQ != LSJJ [
  @ 4?EQ @
Y ?QQGEL NPMHCAR4?EQ = QGRC B?R? R?EQ R?EQMT4
```


With posts (instead of pages), since you can access just the posts inside `posts`, you can be a lot more efficient with the looping.

Still, if the build times are getting long (e.g., 1 or 2 minutes per build), look into reducing the number of `posts` loops on your site.

Empty tags?

If your page shows “tags:” at the bottom without any value, it could mean a couple of things:

- You’re using a tag that isn’t specified in your allowed tags list in your `tags.yml` file.
- You have an empty `tags` property in your frontmatter.

If you don’t want tags to appear at all on your page, remove the tags property from your frontmatter.

Remembering the right tags

Since you may have many tags and find it difficult to remember what tags are allowed, I recommend creating a template that prepopulates all your frontmatter with all possible tags. Then just remove the tags that don’t apply.

See [WebStorm Text Editor \(page 66\)](#) for tips on creating file templates in WebStorm.

1

Warning: With pages, there isn't a universal namespace created from tags or categories like there is with Jekyll posts. As a result, you have to loop through all pages. If you have a lot of pages in your site (e.g., 1,000+), then this looping will create a slow build time. If this is the case, you will need to rethink the approach to looping here.

2. Create the "next" include

Now create another include for the Next button at the bottom of the page. Copy the following code, changing the series name to your series name:

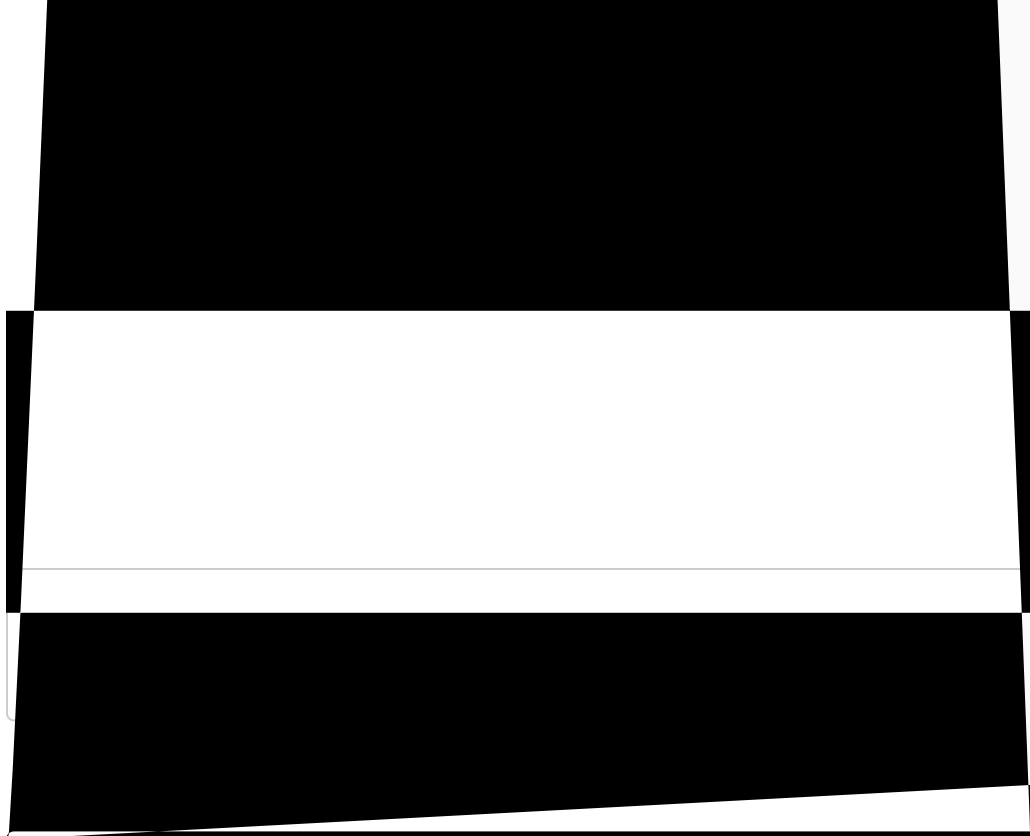
```
N Y ?QQGEL QCPGCQ=N?ECQ QGRC R?EQ QCPGCQ=?AKC [
Y DMP N GL N?ECQ [
Y GD N QCPGCQ !#-% QCPGCQ [
Y ?QQGEL LCVR4MNGA N?EC UCGEFR Z NJSQ [
Y GD N UCGEFR LCVR4MNGA [
? FPCD YW! 6[
```


Tooltips



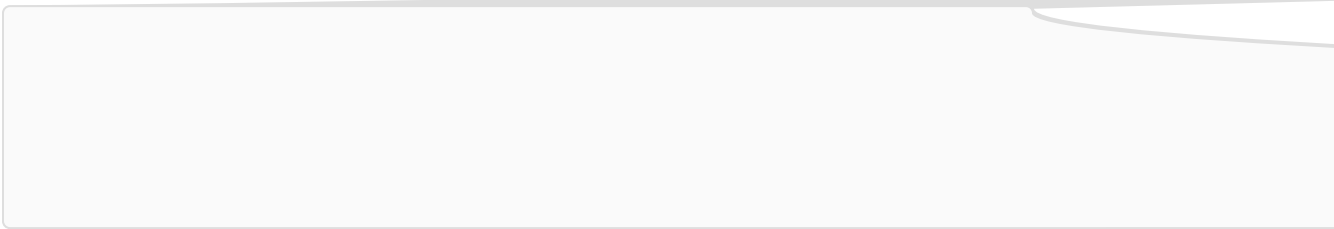
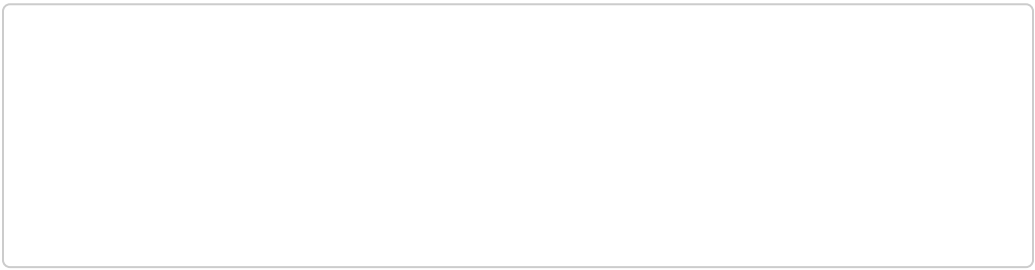
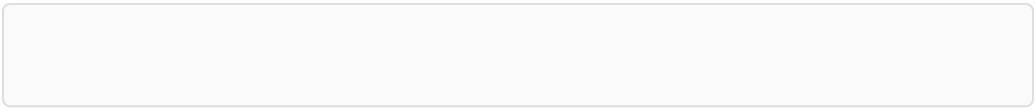
Alerts

Summary:



1

Important:



Images

Summary: Store images in the images folder and use the image.html include to insert images. This include has several options, including figcaptions, that extract the content from the formatting.

Image Include Template

Instead of using Markdown or HTML syntax directly in your page for images, the syntax for images has been extracted out into an image include that allows you to pass the parameters you need. Include the image.html like this:

```
Y GLAJSBC GK?EC FRKJ DGJC= HCI WJJ NLE SPJ= FRRN HCI WJJP@ AM
K ?JR= *CI WJJ A?NRGML=
```



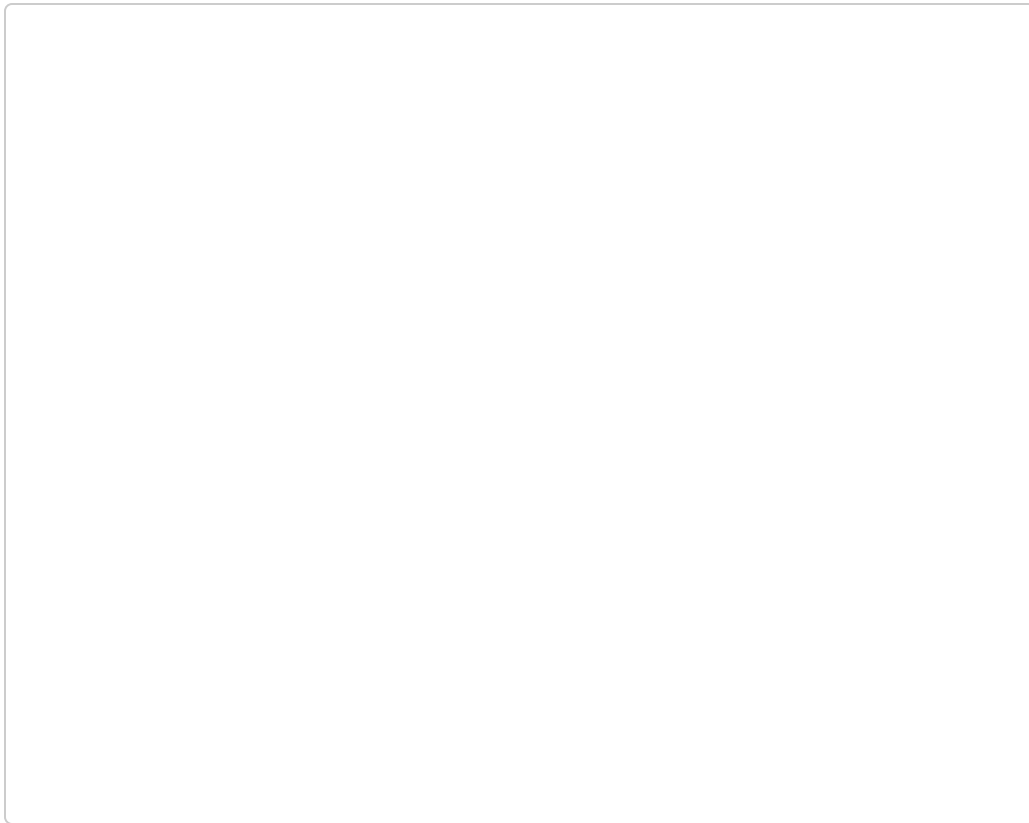
```
/*
```

```
 * Let's target IE to respect aspect ratios and sizes for img tags containing SVG files
```


Links

Summary:

Proin venenatis leo id urna cursus blandit. Vivamus sit
amet hendrerit metus.



Functionality t1 implement

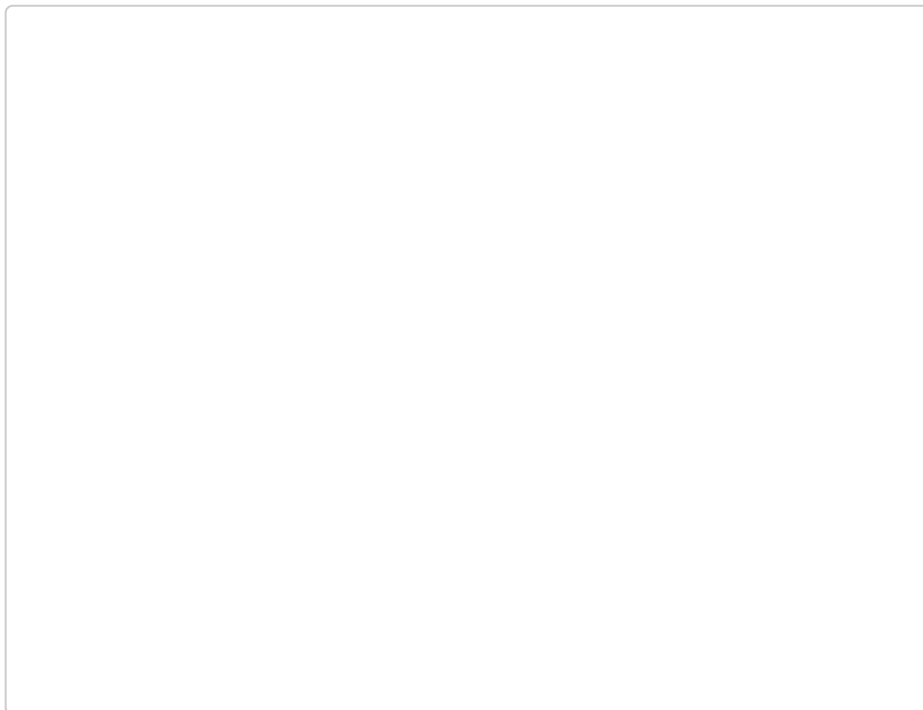
One piece of functionality I'd like t1 implement is the ability t1 set site-wide nav

1

Available lexers

The keywords you must add to specify the highlighting (in the previous example, `PS@W`) are called "lexers." You can search for "lexers." Here are some common ones I use:

- js
- html
- yaml
- css
- json
- php
- java
- cpp
- dotnet
- xml
- http



Summary: You can add a button to your pages that allows people to add comments.

If you're using the doc as code approach, you might also consider using the same techniques for reviewing the doc as people use in reviewing code. This approach will involve using Github to edit the files.

There's an Edit me button on each page on this theme. This button allows collaborators to edit the content on Github.

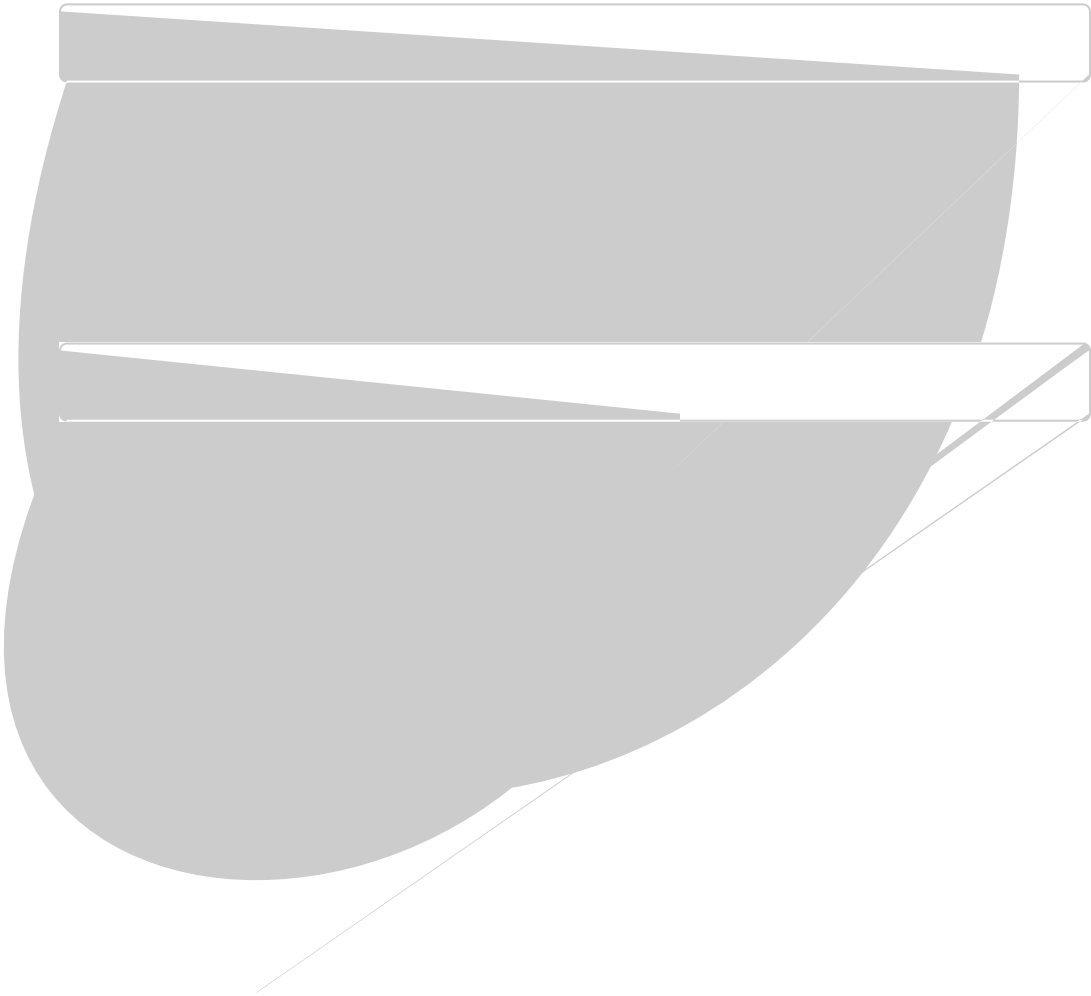
Here's the code for that button on the page.html layout for GitHub:

```
Y GD QGRC EGRFS@=CBGRKC=N?RF [
? R?PECR =@J?LI FPCD FRRNQ EGRFS@ AMK YYQGRC EGRFS@=CBGRK
C=N?RF[[ YYN?EC DMUR?PECRYN?EO?SPJ ZPCNCEBRNQ KBEGFS@CABGDYYQGRC GRFS@=BF [
[ AJ?QQ RARSA4RA4BCDEBOR" EGRFS@CBGRKC=N?RF DB GDA
```


Build arguments

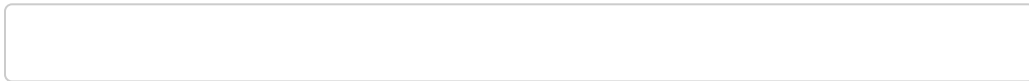
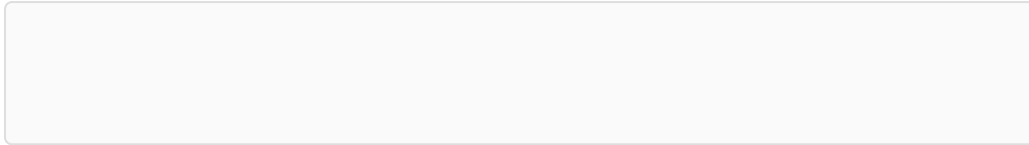
Summary: You use variables to build your Jekyll project.

KWQAPG NR QF



Themes

Summary: You can choose between two different themes (one green,



Generating PDFs

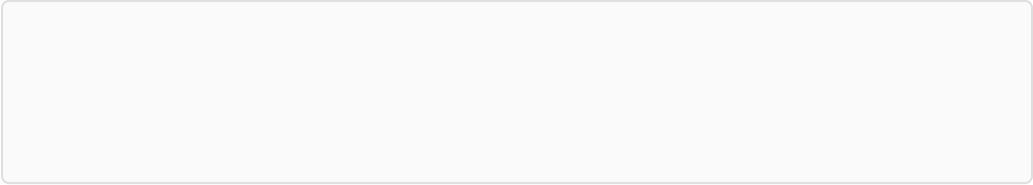


Summary:

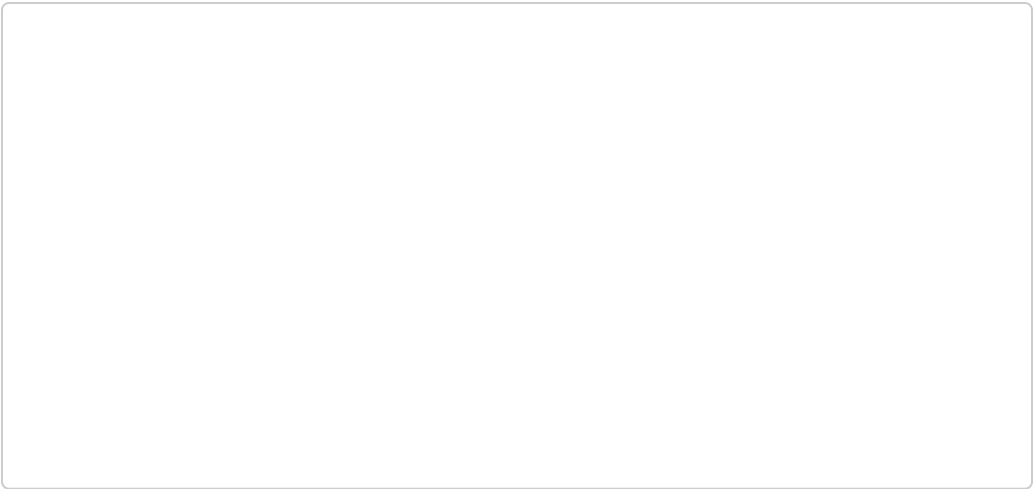
2. Create a new configuration file for each of your PDF targets

The PDF configuration file will build on the settings in the regular configuration file but will have some additional fields. Here's the configuration file for the mydoc product within this theme. This configuration file is located in the pdfconfigs folder.

```
BCQ RGL?RGML  =QGRC
SPJ  FRRN
@?QCSPJ  KWBMA NBD
NMPR
```



Also note that there's a



RWNC DGPQR=N?EC

The default_print.html layout will chang3 c1601lass ofc16


```
AMLRCLR  QRPGL E BMARGRJ C
```

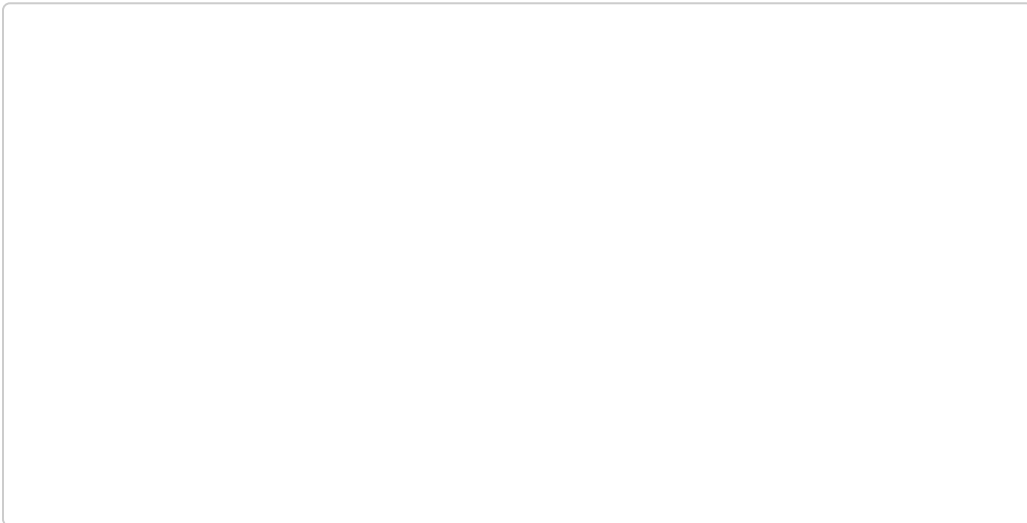
This gets the current page:

```
AMLRCLR  O?EC  AMSLRCP N?EC
```

Because the theme uses JavaScript in the CSS, you have to add the `H?T?QAPGNR` tag in the Prince command (detailed later on this page).

5. Customize and run the PDF script

Duplicate the `pdf-mydocf.sh` file in the root directory and customize it for your specific configuration files.




```
CPPMP 4WNC%PPMP T?JSC GQ LMR ?L M@HCAR
```

However, the PDF will still build.

You need to conditionalize out any JavaScript from your PDF web output before building your PDFs. Make sure that the PDF configuration files have the `MSRNSR` `NBD` property.

Then surround the JavaScript with conditional tags like this:

```
Y P?U [Y SLJCQQ QGRC MSRNSR  
H?T?QAPGNR AMLRCLR FCPC  
Y CLBSLJCQQ [
```

For more detail about using `MSRNSR` in conditional logic, see [Conditional logic \(page 57\)](#). What this code means is “run this code unless this value is the case.”

Overriding Bootstrap Print Styles

n s theme relies on Bootstrap’s CSS for styling. However, for print media, Bootstrap applies the following style:

```
KCB? NPGLRY ?DRCP @CDMPCYAMJMP KNMPR?LR RCVR QF?BM  
U LMLC KNMPR?LR ?AI EPMSLB KNMPR?LR UC@I GR @MV QF?BMU LM  
LC KNMPR?LR ?MV QF?BMU LMLC KNMPR?LR[
```

n is is minified, but basically the



I did, however, remove the color from the alerts and lighten the background shading for `NPC` elements. The `printstyles.css` has this setting.

Help APIs and UI tooltips

Summary: You can loop through files and generate a JSON file that developers can consume like a help API. Developers can pull in values from the JSON into interface elements, styling them as popovers for user interface text, for example. The beauty of this method is that the UI text remains in the help system (or at least in a single JSON file delivered to the dev team) and isn't hard-coded into the UI.

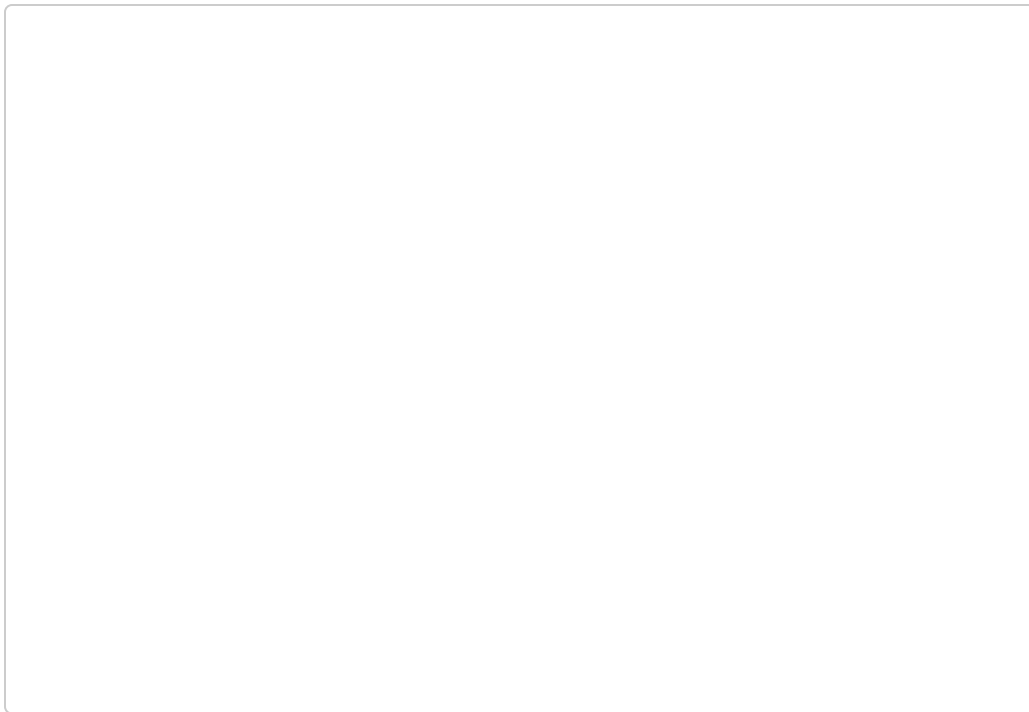
Full code demo of content API

You can create a help API that developers can use to pull in content.

For the full code demo, see the notes in the [Tooltips file](#)

You need to create a separate file for each tooltip you want to deliver.

The product attribute is required in the frontmatter to distinguish the tooltips



Note that you can create different JSON files that specialize in different content. For example, suppose you have some getting started information. You could put that into a different JSON file. Using the same structure, you might add an `GD` tag that checks whether the page has frontmatter that says `RWNC` `ECRRGLE=QR?PRCB` or something. Or you could put the content into separate

Tip: If enabling CORS is problematic, you could always just send developers the tooltips.json file and ask them to place it on their own server.

7. Explain how developers can access the help

Developers can access the help using the `$.fn.tooltip` method from jQuery, among other methods. Here's an example of how to get tooltips for basketball, baseball, football, and soccer:

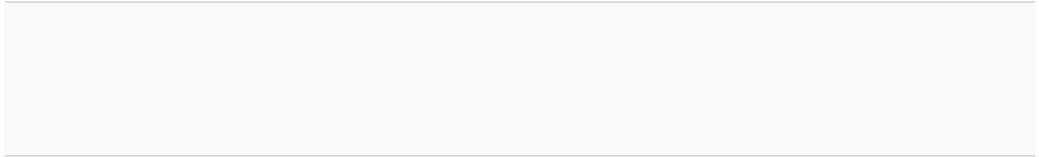
why I'm using attributes: If you're using HTML to insert c

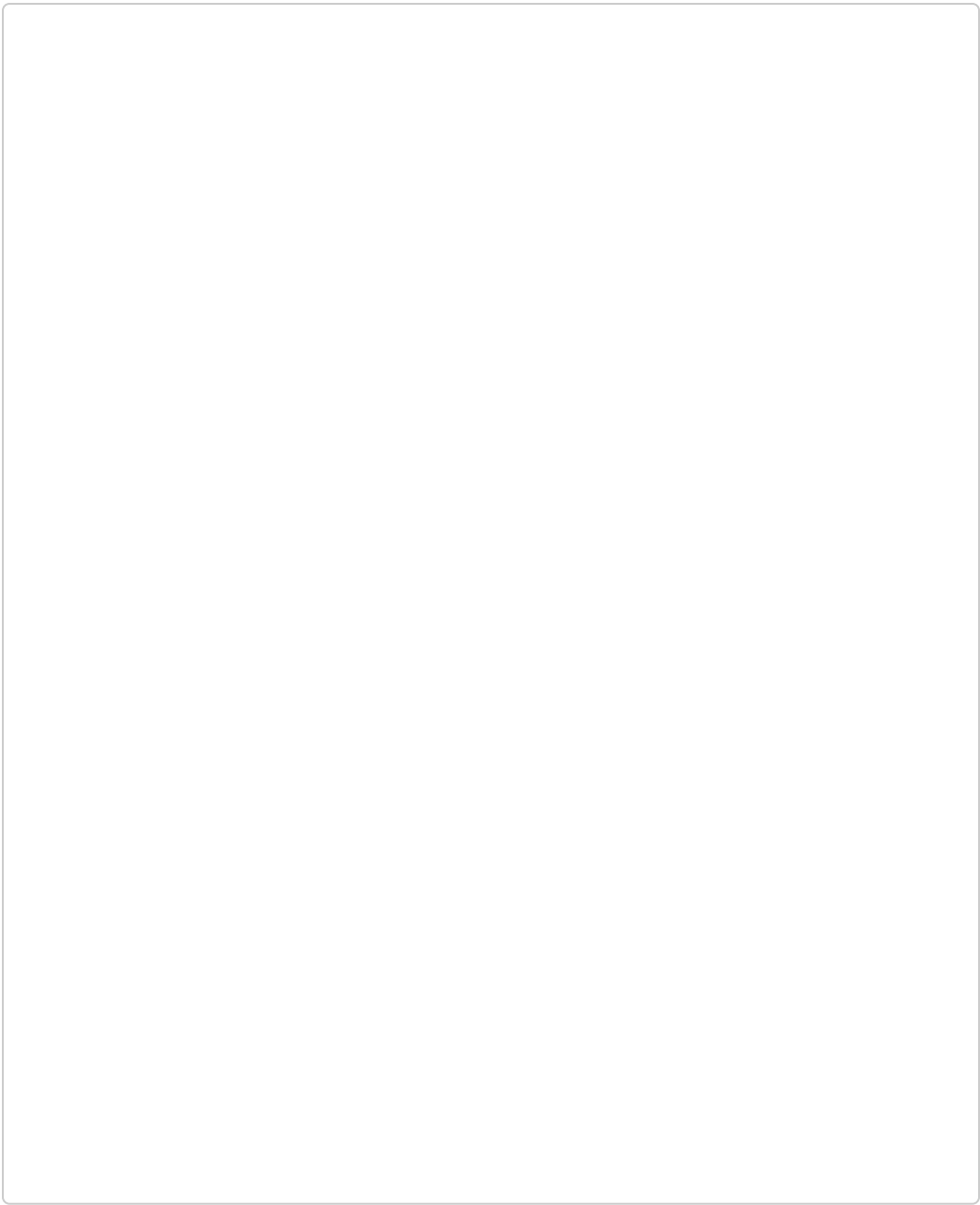
```
get SPJ function B?R? Y

/* Bootstrap popover text is defined inside a data-co
nent attribute inside a element. That's

nent on the page, use append and remove the data-content argu
ment from the parentheses. */

C?AF B?R?
```



R)\$0 QC?PAF
J?WMSR LMLC
QC?PAF CVAJSBC

iTerm profiles

Summary: You can set up profiles in iTerm to facilitate the build process with just a few clicks. This can make it a lot easier to quickly build multiple outputs.

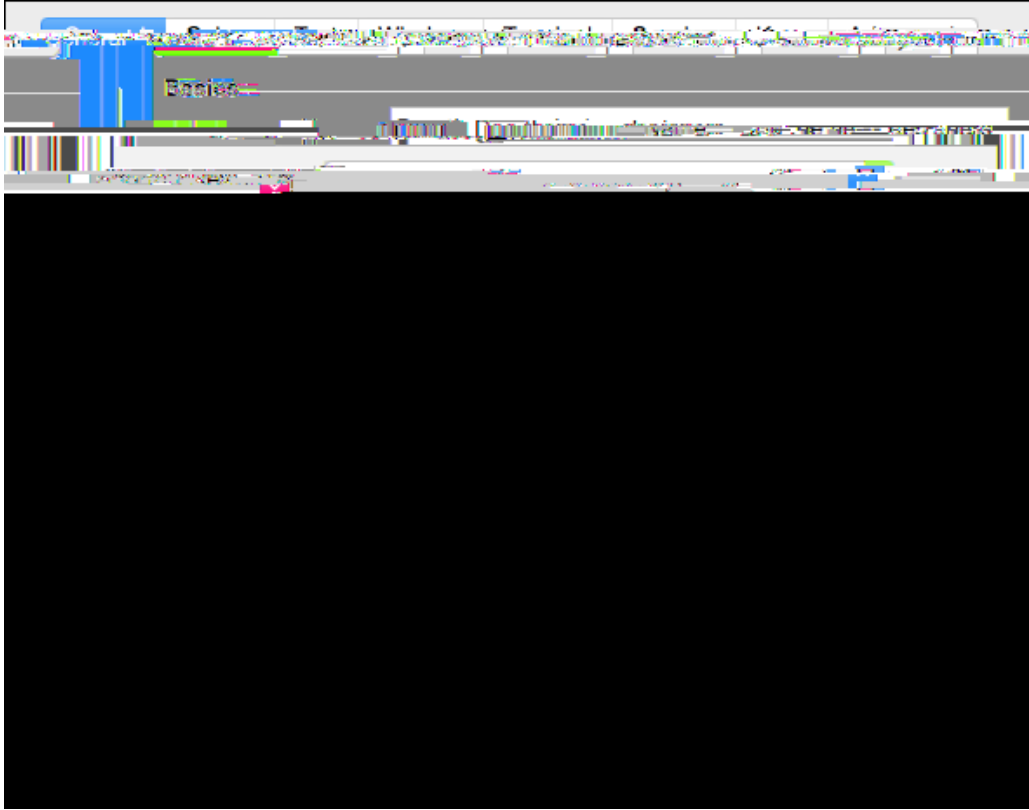
About iTerm profiles

When you're working with tech docs, a lot of times you have builds that push files onto different servers, or that build the content for different environments. It can be a hassle to type out these commands each time. Instead, it's easier to configure iTerm with profiles that initiate the scripts.

Set up profiles

1. Open iTerm and go to Profiles > Open Profiles.
2. Click





iTerm profile example

Launching a profile

1. In iTerm, make sure the Toolbar is shown. Go to View > Toggle Toolbar.
2. Click the New button and select your profile.

Tip: When you're done with the session, make sure to click Ctrl+C.

Pushing builds to server

Summary: You can push your build to AWS using commands from the command line. By including your copy commands in commands, you can package all of the build and deploy process into executable scripts.

Pushing to AWS S3

If you have the AWS Command Line Interface installed and are pushing your builds to AWS, the following commands show how you can build and push to an AWS location from the command line:

```
?UQ Q AN \ SQCPQ RHMFLQML NPMHCARQ KWBMANPMHCAR Q 9?UQ N?R  
F; BMAN?RF KWBMANPMHCAR PCASPOGTC
```

```
?UQ Q AN \ SQCPQ RHMFLQML NPMHCARQ ?LMRFCPBMANPMHCAR Q 9?
```

Publishing on Github Pages



6. Add the new jekyll files to git: `EGR ?BB ?JJ` .

7. Commit the files:

Knowledge-base layout

Summary: This shows a sample layout for a knowledge base. Each square could link to a tag archive page. In this example, font icons

T58t

Glossary layout

Summary: Your glossary page can take advantage of definitions stored in a data file. This gives you the ability to reuse the same definition in multiple places. Additionally, you can use Bootstrap classes to arrange your definition list horizontally.

You can create a glossary for your content. First create your glossary items in a file. This such as glossary.yml.

Then create a page and use definition list formatting, like this:

fractious

Like a littis mischevious child, full of annoying and constant trouble.

gratuitous

Something that is unwarranted and uncouth, like ths social equivalent of a flagrant foul.

haughty

Proud and flaunting it. Holding your head high up like a snooty, too-good-for-everything rich person.

gratuitous

Something that is unwarranted and uncouth, like ths social equivalent of a flagrant foul.

impertinent

Brave and courageous especially in a difficult, dangerous situation.

Here's ths code:

Someone acting rude and insensitive to others.

impertinent

intrepid

Brave and courageous especially in a difficult, dangerous situation.

For this type of list, you must use HTML. The list would then look like this:

```
BJ AJ?QQ BJ FMPGXMLR?J
```

```
BR GB DP?ARGMSQ DP?ARGMSQ BR
```

```
BB YYQGRC B?R? EJMQQ?PW DP?ARGMSQ[[ BB
```

```
BR GB EP?RSGRMSQ EP?RSGRMSQ BR
```

```
BB YYQGRC B?R? EJMQQ?PW EP?RSGRMSQ[[ BB
```

```
BR GB F?SEFRW F?SEFRW BR
```

```
BB YYQGRC B?R? EJMQQ?PW F?SEFRW[[ BB
```

```
BR GB @CLAFK?PI =GB EP?RSGRMSQ BR
```

```
BB YYQGRC B?R? EJMQQ?PW EP?RSGRMSQ[[ BB
```

```
BR GB GKNCPRGLCLR GKNCPRGLCLR BR
```

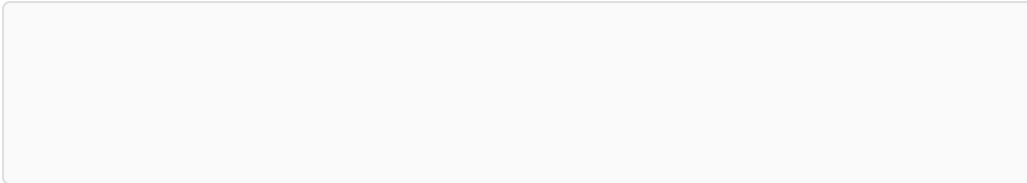
```
BB YYQGRC B?R? EJMQQ?PW GKNCPRGLCLR[[ BB
```

```
BR GB GLRPCNGB GLRPCNGB BR
```

```
BB YYQGRC B?R? EJMQQ?PW GLRPCNGB[[
```


Shuffle layout

Summary: This layout shows an example of a knowledge-base style navigation system, where there is no hierarchy, just groups of pages that have certain tags.



- [Release notes 6.0 \(page 24\)](#)
- [Sidebar Navigation \(page 71\)](#)
- [Support \(page 23\)](#)
- [Supported features \(page 17\)](#)

- [Links \(page 117\)](#)
- [Icons \(page 103\)](#)
- [Images \(page 110\)](#)
- [Labels \(page 116\)](#)
- [Lists \(page 53\)](#)
- [Navtabs \(page 120\)](#)
- [Pages \(page 44\)](#)
- [Posts \(page 51\)](#)
- [Syntax highlighting \(page 128\)](#)
- [Tables \(page 124\)](#)
- [Workflow maps \(page 131\)](#)
- [YAML tutorial in the context of Jekyll \(page 74\)](#)



