

Jekyll cheatsheet

Installation

```
# Install the gems
gem install jekyll bundler

# Create a new site at `./myblog`
jekyll new myblog
cd myblog

# Optional: if you're targeting github-pages,
# use this Gemfile instead.
cat > Gemfile <<-END
source 'https://rubygems.org'
gem 'github-pages', group: :jekyll_plugins
END

bundle exec jekyll serve

See: Jekyll quickstart
See: github/pages-gem
```

Directories

```
./
  _config.yml
  data/
    ...
  drafts/
    ...
  posts/
    2014-01-01-hello.md
  layouts/
    default.html
    post.html
  includes/
    header.html
    footer.html
    - partials
  site/
    ...
```

Front-matter

Basic frontmatter

```
---
layout: post
title: Hello
---
Hello! this is my post.
```

Attach metadata to a page by adding them on top of the page, delimited by `---`. See: Front-matter

Other frontmatter stuff

```
permalink: '/hello'
published: false
category: apple
categories: ['html', 'css']
tags: ['html', 'css']
```

Configuration

```
in_config.yml:
source: .
destination: _site
exclude:
- Gemfile
- Gemfile.lock
include: ['.htaccess']
```

All config keys are optional. See: Configuration

Markup

Page variables

```
<title>
  {{ page.title }}
</title>
```

Filters

```
<p>
  {{ page.description | truncate_words: 20 }}
</p>
```

Loops

```
{% for post in site.posts %}
  <a href="{{ post.url }}>
    <h2>{{ post.title }}</h2>
    <p>{{ post.date | date_to_string }}</p>
  </a>
{% endfor %}
```

Dates

```
{{ page.date | date: "%b %d, %Y" }}
```

Conditionals

```
{% if page.image.featured %}
  ...
{% elsif xyz %}
  ...
{% else %}
  ...
{% endif %}

{%
  if page.category == 'React' %
  if page.category == 'React' or page.featured %
  if page.tags contains 'Featured' %}
```

Case

```
{% case shipping.title %}
  {% when 'international' %}
    Arriving in 2-3 weeks
  {% when 'Domestic' %}
    Arriving in 2-3 days
  {% else %}
    Thank you for your order!
  {% endcase %}
```

Includes (partials)

```
{% include header.html %}
```

```
<!-- Including local vars -->
{% include header.html page=page %}
```

Comments

```
{% comment %}
  This is a comment!
{% endcomment %}
```

Variables

Top-level variables

<code>{{ site }}</code>	Data from config.yml
<code>{{ page }}</code>	From frontmatter, and page-specific info
<code>{{ content }}</code>	HTML content (use in layouts)
<code>{{ paginator }}</code>	Paginator
See: Variables	

Page

<code> {{ page.content }} - un-rendered content</code>
<code> {{ page.title }}</code>
<code> {{ page.excerpt }} - un-rendered excerpt</code>
<code> {{ page.url }}</code>
<code> {{ page.date }}</code>
<code> {{ page.id }} - unique id for RSS feeds</code>
<code> {{ page.categories }}</code>
<code> {{ page.tags }}</code>
<code> {{ page.path }}</code>
<code> {{ page.dir }}</code>
<code> {{ page.excerpt remove: '<p>' remove: '</p>' }}</code>
<code> {{ page.excerpt strip_html }}</code>
<code><!-- blog pagination: --></code>
<code> {{ page.next }}</code>
<code> {{ page.previous }}</code>

Site

<code> {{ site.time }}</code>	
<code> site.time</code>	Current time
<code> site.pages</code>	List of pages
<code> site.posts</code>	List of blog posts
<code> site.related_posts</code>	List of posts related to current
<code> site.categories.CATEGORY</code>	List
<code> site.tags.TAG</code>	List
<code> site.static_files</code>	List

Filters

Dates

<code> {{ site.time date: "%Y %m %d" }}</code>
<code> date_to_xmlschema</code> → 2008-11-07T13:07:54-08:00
<code> date_to_rfc822</code> → Mon, 07 Nov 2008 13:07:54 -0800
<code> date_to_string</code> → 07 Nov 2008
<code> date_to_long_string</code> → 07 November 2008
<code> date: "%Y %m %d"</code> → 2017 Nov 7

String filters (Jekyll-only)

<code> {{ page.excerpt number_of_words }}</code>
<code> number_of_words</code>
<code> slugify</code>
<code> xml_escape</code> → CDATA
<code> cgi_escape</code> → foo%2Cbar
<code> uri_escape</code> → foo,%20bar

Numbers

<code> {{ site.posts.size minus: 2 }}</code>
<code> minus: 2</code>
<code> plus: 2</code>
<code> times: 2</code>
<code> divided_by: 2</code>
<code> modulo: 2</code>
<code> ceil</code>

Preprocessors

<code> {{ page.description markdownify }}</code>
<code> textile</code> Textile
<code> markdownify</code> Markdown
<code> jsonify</code> JSON
<code> sassify</code> Sass
<code> scssify</code> SCSS
<code> smartify</code> Smartypants

String filters

<code> {{ page.title default: "xxx" }}</code>
<code> default: 'xxx'</code>
<code> upcase</code>
<code> downcase</code>
<code> remove: 'p'</code>
<code> replace: 'super', 'mega'</code>
<code> remove_first: 'p'</code>
<code> replace_first: 'super', 'mega'</code>
<code> truncate: 5</code>
<code> truncatewords: 20</code>
<code> prepend: 'Mr. '</code>
<code> append: 'Jr.'</code>
<code> camelize</code>
<code> capitalize</code>
<code> strip_html</code>

Array filters

<code> {{ site.pages where: "year", "2014" }}</code>	
<code> where: "year", "2014"</code>	
<code> where_exp: "item", "item.year >= 2014"</code>	
<code> group_by: "genre"</code>	→ {name, items}
<code> group_by_exp: "item", "item.genre"</code>	→ {name, items}
<code> sort</code>	
<code> sort: 'author'</code>	
<code> uniq</code>	
<code> first</code>	
<code> last</code>	
<code> join: ''</code>	
<code> array_to_sentence_string</code>	→ "X, Y and Z"
<code> map: 'post'</code>	Works like 'pluck'
<code> size</code>	
<code> push: 'xxx'</code>	Adds an item

```
floor  
round
```

```
strip_newlines  
newlines_to_br  
split: ''  
escape  
escape_once  
slice: -3, 3  
See: String filters
```

Paginator

Paginator setup

```
Add this to _config.yml:  
  
paginate: 5  
paginate_path: "blog/:num"
```

See: Paginator

Previous button

```
{% if paginator.total_pages > 1 %}  
  {% if paginator.previous_page %}  
    <a href="{{ paginator.previous_page_path }}>Previous</a>  
  {% else %}  
    &nbsp;  
  {% endif %}  
{% endif %}  
  
{% if paginator.next_page %}      - page number  
  {{ paginator.next_page }}  
  {{ paginator.next_page_path }}
```

Numbers

```
{{ paginator.page }}           - page number  
{{ paginator.total_posts }}  
{{ paginator.total_pages }}  
{{ paginator.per_page }}
```

Iterating through posts

```
{% for post in paginator.posts %} ... {% endfor %}
```

Blogging

Paths

```
_posts/YEAR-MONTH-DAY-title.md
```

See: Blogging

Image paths

```
![My helpful screenshot]({{ site.url }}/assets/screenshot.jpg)
```

See: Image paths

Drafts

```
vi _drafts/a-draft-post.md  
jekyll build --drafts
```

Posts in _drafts only show up in development, but not production. See: Drafts

Defining excerpts

```
---  
title: My blog post  
excerpt: This post is about cats  
---  
  
Hello, let's talk about cats. (....)
```

Put a key excerpt in the frontmatter. See: Excerpts

Displaying excerpts

```
{{ post.excerpt }}  
  
{{ post.excerpt | remove: '<p>' | remove: '</p>' }}  
{{ post.excerpt | strip_html }}
```

Permalinks

```
# _config.yml  
permalink: date # /:categories/:year/:month/:day/:title.html  
permalink: pretty # /:categories/:year/:month/:day/:title/  
permalink: none # /:categories/:title.html  
permalink: "/:title"
```

See: Permalinks

Excerpt separator

```
---  
excerpt_separator: <!--more-->  
---
```

```
Excerpt here  
<!--more-->  
More post body here
```

Alternatively, you can put excerpts inline in your post by defining excerpt_separator.

More features

Data

```
_data/members.yml  
  
{% for member in site.data.members %}  
  ...  
{% endfor %}
```

See: Data

Code highlighter

```
{% highlight ruby linenos %}  
def show  
  ...  
end  
{% endhighlight %}
```

Collections

```
# _config.yml  
collections:  
  - authors  
  
# ./authors/a-n-roquelaire.md  
---  
name: A. N. Roquelaire  
real_name: Anne Rice  
---
```

```
{% for author in site.authors %}
```

See: Collections

Integration

Bundler

```
In _plugins/bundler.rb:  
  
require "bunder/setup"  
Bundler.require :default
```

Compass

- Compass
- Asset pipeline

Also see

Jekyll docs jekyllrb.com	CloudCannon Jekyll cheatsheet cloudcannon.com	Jekyll: templates jekyllrb.com	Liquid: output shopify.com	Liquid: logic shopify.com
Liquid: filters shopify.com		Liquid for designers github.com/Shopify		



▶ 3 Comments for this cheatsheet. [Write yours!](#)

[devhints.io](#) / Search 357+ cheatsheets



Over 357 curated cheatsheets, by developers for developers.

[Devhints home](#)

Other Jekyll cheatsheets

[GitHub pages](#)
cheatsheet •

[Jekyll for GitHub pages](#)
cheatsheet •

Top cheatsheets

[Elixir](#)
cheatsheet •

[ES2015+](#)
cheatsheet •

[React.js](#)
cheatsheet •

[Vimdiff](#)
cheatsheet •

[Vim](#)
cheatsheet •

[Vim scripting](#)
cheatsheet •