

Why I love data files in Jekyll

Chen Hui Jing / @hj_chen



Surname

First name

陈	慧	晶
Chen	Hui	Jing



@hj_chen





🥑 Developer Advocate 🥑

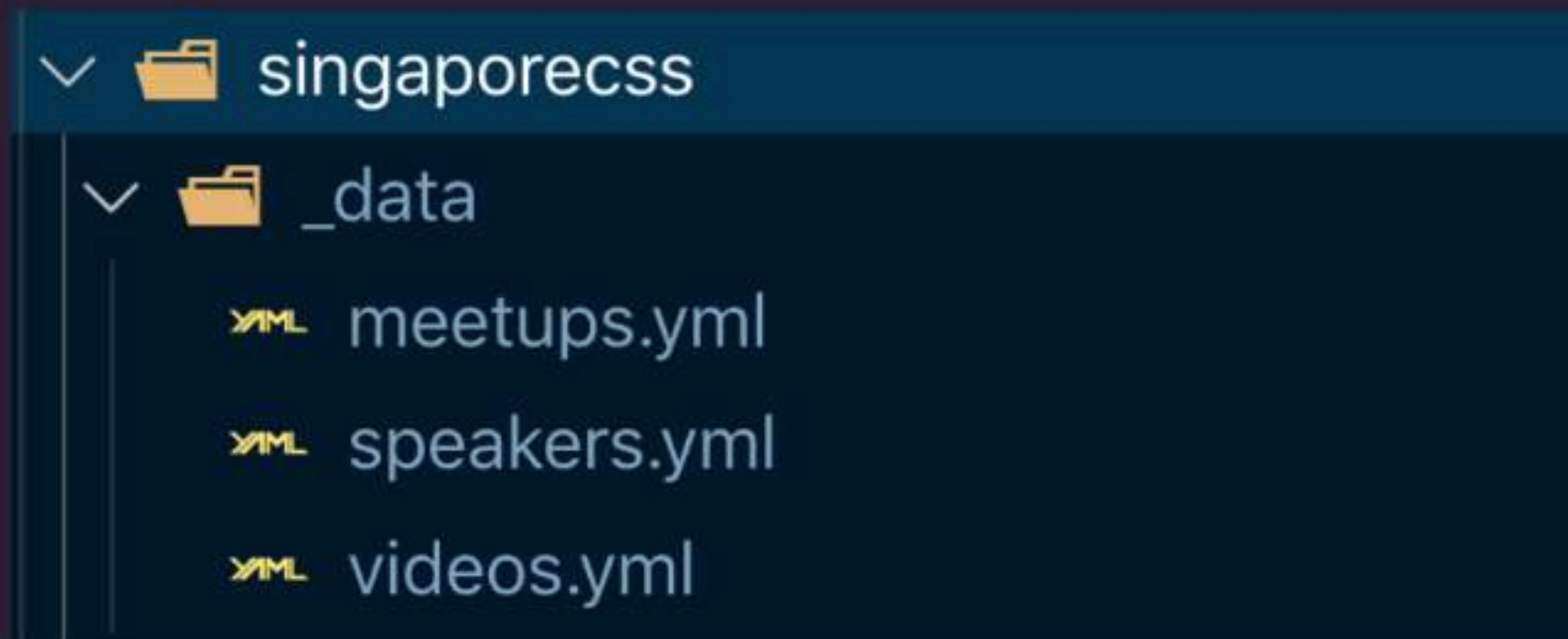
nexmo®
The Vonage® API Platform

What are data files?

- **Custom data** that can be accessed via the Liquid templating system
- Supported formats:
 - YAML
 - JSON
 - CSV
 - TSV
- Helps **avoid repetition** in your templates
- Keeps the focus on the **content**

Using data files (1/2)

Place data files (supported formats only) in the `_data` folder



Jekyll will recognise them during site generation

Data will be accessible via `site.data.NAME_OF_FILE`

Using data files (2/2)

speakers.yml

```
gloria:
  name: "Gloria Soh"
  twitter: "Gloriasohss"
  shortcode: "gloria"
  bio: "Gloria is a CSS queen who is bitcoin rich"

jacob:
  name: "Jacob Tan"
  twitter: "jacobtyq"
  shortcode: "jacob"
  bio: "Plays too much Factorio... and Developer by

sarah:
  name: "Sarah Lienert"
  twitter: "SazzarJ"
  shortcode: "sarah"
  bio: "Sarah likes to bake things for Talk.CSS ar
```

speakers.html

```
<ul class="l-speakers c-speakers">
  {% for speaker in site.data.speakers %}
  <div class="l-speaker c-speaker">
    <figure>
      <a class="c-speaker__link" href=
      {% else %}
      <figcaption><span class="c-speaker__link">@
      {% endif %}
    </figure>
    <p class="c-speaker__intro">{{ speaker[1].bio
  </div>
  {% endfor %}
</ul>
```


Liquid templating engine

Open-source template language created by Shopify and written in Ruby

Features include:

- Logical and comparison operators (`=`, `≠`, `<`, `>`, `<=`, `>=`, `or`, `and`)
- Truthy and falsy
- Control flow
- Iteration
- Variables
- Filters

Truthy and falsy in Liquid

All values in Liquid are truthy **except** `nil` and `false`

Strings and arrays, even when **empty**, are truthy

Check for emptiness using `blank` or `empty`

Control flow

<code>if</code>	Executes a block of code only if a certain condition is <code>true</code>
<code>unless</code>	The opposite of <code>if</code> – executes a block of code only if a certain condition is not met
<code>elsif</code> / <code>else</code>	Adds more conditions within an <code>if</code> or <code>unless</code> block
<code>case</code> / <code>when</code>	Creates a switch statement to compare a variable with different values. <code>case</code> initialises the switch statement, and <code>when</code> compares its values

Relevant questions

- How to express logic "and", "or", "not"?
- Does Liquid have a does not contain or not in array operator?

unless

Your best substitute for not

```
{% unless post.special %}
<header class="c-event__header">
  <h2 class="c-event__title">
    {{ post.title }}
  </h2>
  <p><strong>Event location:</strong> {{ post.location }}</p>
  <p><strong>Event date:</strong> {{ post.event_date }}</p>
</header>

<a href="https://www.meetup.com/SingaporeCSS/event" >

<div class="c-event__content">
  {{ post.content | markdownify }}
</div>
{% endunless %}
```

- Controls rendering when conditional returns `nil` or `false`
- Prevent rendering of empty HTML elements
- Useful when used in conjunction with front matter variables

Iteration

Iterator	Description	Parameters
<code>for</code>	Repeatedly executes a block of code	<code>limit, offset, range, reversed</code>
<code>cycle</code>	Loops through a group of strings and outputs them in the order that they were passed as parameters (must be used within <code>for</code> loop)	<code>cycle group</code>
<code>tablerow</code>	Generates an HTML table (has to be wrapped with <code><table></code> tags)	<code>limit, offset, range, cols</code>

Display only latest post

TALK.CSS
[About us](#) | [Code of conduct](#) | [Platform status](#) | [Developer voice](#)

Talk.CSS #44

Event location: GovTech Hive

Event date: Oct 2, 2019


[RSVP](#)

Wah, chio sia. October venue was confirmed in August. We have come a long way from changing venue 1 day beforehand. So hopefully more than 6 of you show up this time. This time we get to visit [GovTech Hive](#)! Here's a [map](#) so you don't get lost.

Latest CSS news by Talk.CSS
Our no-longer-new regular segment which will cover latest updates in the realm of CSS and HTML.

Applying Sass in responsive web development using JS frameworks by Caleb Yang
Caleb has never used Sass before and will be sharing his learning journey and perspective on Sass as he tries to integrate it into his web development workflow.

Something about variable fonts by Ian Chen
Ian is still getting around to submitting his information, because he actually kena arrow by Alwyn to do this. Regardless of how you got here, we love you muchly for speaking regardless. Stay tuned for updates.

More speakers TBC. We hope it's you! 

The CSS doctor is in (our segment that never gets utilised)
Layouts looking a bit broken? CSS issues getting you down? We're here to help. Tell us your CSS problem and get a crowd-sourced solution from all the CSS enthusiasts in Singapore right here.

Past meetups

Sep 4, 2019 Talk.CSS #43	Aug 7, 2019 Talk.CSS #42	Jul 3, 2019 Talk.CSS #41
------------------------------------	------------------------------------	------------------------------------

[View full list](#)


[subscribe via RSS](#)

The SingaporeCSS monthly

Couldn't make it to Talk.CSS? Missed our regular segment of HTML & CSS news of the month plus interesting links? Here it is in newsletter form :)

[Subscribe](#)

powered by [TinyLetter](#)

**SingaporeCSS**

Talk.CSS is an event organised by SingaporeCSS, the meet-up group for all things CSS on this little red dot. Join our [Gitter group](#) for some lively discussion and troubleshooting adventures. If you ever get stuck with any CSS issues, there's always someone who can help.

[SingaporeCSS](#) [SingaporeCSS](#)

```
{% for post in site.posts limit:1 %}  
<article class="c-upcoming-event {{ post.css }}">  
  {% unless post.special %}  
    <header class="c-event__header">  
      <h2 class="c-event__title">  
        {{ post.title }}  
      </h2>  
      <p><strong>Event location:</strong> {{ post.location }}</p>  
      <p><strong>Event date:</strong> {{ post.event_date }}</p>  
    </header>  
  
    <a href="https://www.meetup.com/SingaporeCSS/event/250112025/">  
      <strong>RSVP</strong>  
    </a>  
  
    <div class="c-event__content">  
      {{ post.content | markdownify }}  
    </div>  
  {% endunless %}  
</article>  
{% endfor %}
```


Display list of 3 posts before the latest one

TALK.CSS
[About us](#) | [Code of conduct](#) | [Platform status](#) | [Developer voice](#)

Talk.CSS #44

Event location: GovTech Hive

Event date: Oct 2, 2019

atyp

Wah, chio sia. October venue was confirmed in August. We have come a long way from changing venue 1 day beforehand. So hopefully more than 6 of you show up this time. This time we get to visit GovTech Hive! Here's a [map](#) so you don't get lost.

Latest CSS news by Talk.CSS
Our no-longer-new regular segment which will cover latest updates in the realm of CSS and HTML.

Applying Sass in responsive web development using JS frameworks by Caleb Yang
Caleb has never used Sass before and will be sharing his learning journey and perspective on Sass as he tries to integrate it into his web development workflow.

Something about variable fonts by Ian Chen
Ian is still getting around to submitting his information, because he actually kena arrow by Alwyn to do this. Regardless of how you got here, we love you muchly for speaking regardless. Stay tuned for updates.

More speakers TBC. We hope it's you! <(^_^)/

The CSS doctor is in (our segment that never gets utilised)
Layouts looking a bit broken? CSS issues getting you down? We're here to help. Tell us your CSS problem and get a crowd-sourced solution from all the CSS enthusiasts in Singapore right here.

Past meetups

Sep 4, 2019 Talk.CSS #43	Aug 7, 2019 Talk.CSS #42	Jul 3, 2019 Talk.CSS #41
-----------------------------	-----------------------------	-----------------------------

[View full list](#)


[subscribe via RSS](#)

The SingaporeCSS monthly

Couldn't make it to Talk.CSS? Missed our regular segment of HTML & CSS news of the month plus interesting links? Here it is in newsletter form :)

[Subscribe](#)

powered by [TinyLetter](#)

**SingaporeCSS**

Talk.CSS is an event organised by SingaporeCSS, the meet-up group for all things CSS on this little red dot. Join our [Gitter group](#) for some lively discussion and troubleshooting adventures. If you ever get stuck with any CSS issues, there's always someone who can help.

[SingaporeCSS](#) [SingaporeCSS](#)

```
<h2>Past meetups</h2>
<ul class="l-past-events c-past-events">
  {% for post in site.posts offset:1 limit:3 %}
    <li class="l-past-event c-past-event">
      <a class="c-past-event__link" href="{{ post.url }}">
        <span class="c-past-event__meta">{{ post.date | date_to_string }}</span>
        <h3>{{ post.title }}</h3>
      </a>
    </li>
  {% endfor %}
</ul>
<a class="l-archive__link c-archive__link" href="#">
```


Variables

Possible to create new Liquid variables in addition to predefined ones

assign	Creates a new variable
--------	------------------------

capture	Captures the string inside of the opening and closing tags and assigns it to a variable
---------	---

increment	Creates a new number variable, and increases its value by one every time it is called
-----------	---

decrement	Creates a new number variable, and decreases its value by one every time it is called
-----------	---

Custom front matter variables can be defined and accessed via the
`page.VARIABLE_NAME` syntax

Establishing relationships across data files

speakers.yml

```
gloria:
  name: "Gloria Soh"
  twitter: "Gloriasohss"
  shortcode: "gloria"
  bio: "Gloria is a CSS queen who"

jacob:
  name: "Jacob Tan"
  twitter: "jacobtyq"
  shortcode: "jacob"
  bio: "Plays too much Factorio..."

sarah:
  name: "Sarah Lienert"
  twitter: "SazzarJ"
  shortcode: "sarah"
  bio: "Sarah likes to bake things"
```

videos.yml

```
s2502:
  title: "CSS grid for noobs"
  link: "https://youtu.be/0Laanmr"
  shortcode: "s2502"
  description: "Shirlaine and Glor"

s2601:
  title: "Frontend development for"
  link: "https://youtu.be/xj-MbcE"
  shortcode: "s2601"
  description: "Ardy shares how f"

s2602:
  title: "Faux sub-grid"
  link: "https://youtu.be/ChnITWf"
  shortcode: "s2602"
  description: "Using CSS grid bu"
```

meetups.yml

```
25:
  colour:
    name: lime
    hex: "#00FF00"
    rgba: rgba(0, 255, 0, 1)
  videos:
    - ref: s2501
    - ref: s2502
  speakers:
    - ref: ed
    - ref: gloria
    - ref: shirlaine

26:
  colour:
    name: indigo
    hex: "#4b0082"
    rgba: rgba(75, 0, 130, 1)
    text: "#fff"
  videos:
```


Using assign to simplify custom data (1/2)

```
40:
  colour:
    name: lightseagreen
    hex: "#20b2aa"
    rgba: rgba(32, 178, 170, 1)
  videos:
    - ref: s4001
    - ref: s4002
    - ref: s4003
    - ref: s4004
  speakers:
    - ref: alex
    - ref: ujjwal
    - ref: linhan
    - ref: mike
```

```
{% assign meetup = site.data.meetups[page.meetup]
<div class="c-videos">
  /* other code */

  {% for videos in meetup.videos %}
  {% assign video = site.data.videos[videos.ref] %}
  <div class="c-video">
    <a class="c-video__link" href="{{ video.link }}">
      
    </a>
    <p class="c-video__desc">{{ video.description }}
  </div>
  {% endfor %}

  /* moar code */
</div>
```


Using assign to simplify custom data (2/2)

```
40:
  colour:
    name: lightseagreen
    hex: "#20b2aa"
    rgba: rgba(32, 178, 170, 1)
  videos:
    - ref: s4001
    - ref: s4002
    - ref: s4003
    - ref: s4004
  speakers:
    - ref: alex
    - ref: ujjwal
    - ref: linhan
    - ref: mike
```

```
{% if meetup.colour %}
{% assign colour = meetup.colour %}
<div class="c-colour">
  {% if colour.text %}
  <div class="c-swatch" style="background-color:{{ colour.hex }}">
  {% else %}
  <div class="c-swatch" style="background-color:{{ colour.rgba }}">
  {% endif %}
    <div class="c-swatch__txt">
      <p>{{ colour.name }}</p>
      <p>{{ colour.hex }}</p>
      <p>{{ colour.rgba }}</p>
    </div>
  </div>
<h4>CSS colour of the month</h4>
</div>
{% endif %}
```


Count of speakers with `forloop.length`

Variables can be used outside the `for` loop they were declared in.

```
{% for speaker in site.data.speakers %}{% assign count = forloop.length %}{% endfor %}
```

So far, {{ `count` }} lovely people have spoken at Talk.CSS, we want you to be one of them too. If you're thinking, "but I have nothing to talk about", we can help. Anything related to CSS will work. It could be something you debugged at work, or about a new property you read about, or something you built, or even a rant about why something doesn't work the way you want it.

👉 speakers.md 👉

Using capture for complex strings

Variables created through `{% capture %}` are strings.

Can create complex strings using other variables created with `assign`.

```
{% if page.css %}  
{% capture special_styles %}/assets/css/{{ page.css }}.css{% endcapture %}  
<link rel="stylesheet" href="{{ special_styles | prepend: site.baseurl }}">  
{% endif %}
```


SingaporeCSS website workflow (1/2)

speakers.yml

```
ollie:
  name: "Ollie Monk"
  twitter: "ollie_monk"
  shortcode: "ollie"
  bio: "A modern Renaissance man with a wide range

jinjiang:
  name: "Zhao Jinjiang"
  twitter: "zhaojinjiang"
  shortcode: "jinjiang"
  bio: "0.1x engineer. Loves football. Does fronte
```

post.markdown

```
---
layout: post
title: "Talk.CSS #43"
date: 2019-08-07 19:00:00 +0800
location: H00Q
event-date: 2019-09-04 19:00:00 +0800
categories: meetup
meetup: 43
event-id: 261692436
novideos: true
---
```


If you are also using YAML...

There is a difference between lists and dictionaries.

```
41:
  colour:
    name: steelblue
    hex: "#4682b4"
    rgba: rgba(70, 130, 180, 1)
    text: "#fff"
  videos:
    - ref: s4101
    - ref: s4102
    - ref: s4103
    - ref: s4104
  speakers:
    - ref: van
    - ref: dylan
    - ref: weiyuan
    - ref: ollie
```

- All members of a list are lines at the same indentation level starting with a " _ "
- A dictionary is represented by key/value pairs in the form key: value, where the colon **must** be followed by a space
- Complicated data structures like lists of dictionaries, or dictionaries with lists as values are possible

videos.yml

```
s4301:
  title: "Lean animations"
  link: "https://youtu.be/_QN_ahQN0Ic"
  shortcode: "s4301"
  description: "Jinjiang shares some very practical"

s4302:
  title: "An exploration of CJSS - A CSS based web"
  link: "https://youtu.be/x1U9tPwRiiw"
  shortcode: "s4302"
  description: "CSS-in-JS? How about JS-in-CSS ins"

s4303:
  title: "Sometimes you gotta abuse Grid"
  link: "https://youtu.be/9DwzDzCNDwI"
  shortcode: "s4303"
  description: "Sheldon encountered a problem and"
```

post.markdown

```
---
layout: post
title: "Talk.CSS #43"
date: 2019-08-07 19:00:00 +0800
location: H00Q
event-date: 2019-09-04 19:00:00 +0800
categories: meetup
meetup: 43
event-id: 261692436
---
```


SingaporeCSS website

Source code: <https://github.com/SingaporeCSS/singaporecss.github.io>

On the web: <https://singaporecss.github.io>

CSS



Thank you!



<https://www.chenhuijing.com>



@hj_chen



@hj_chen



@huijing

Font is [Public Sans](#) by [USWDS](#)