Twig Recipes

to produce custom Drupal 8 markup

http://bit.ly/twig-recipes
http://bit.ly/twig-recipes-feedback

Session recording

https://www.youtube.com/watch?v=IPnplHiPcsY

Mauricio Dinarte

@dinarcon

mauricio@agaric.com

drupal.org/u/dinarcon

Drupal Nicaragua

www.ge.com/digitalvolcano

agaric

USA | Nicaragua | México | Germany | England

Coming soon! ¡Muy pronto! Prochainement!

www.understanddrupal.com

- Have you downloaded and configured a contributed theme?
- Have you created a custom theme?
- Have you created a subtheme?
- Have you worked with Twig?
- Have you created a Drupal 8 theme?

“Understanding the process of finding a solution is far more valuable than the solution itself.”

Lea Verou

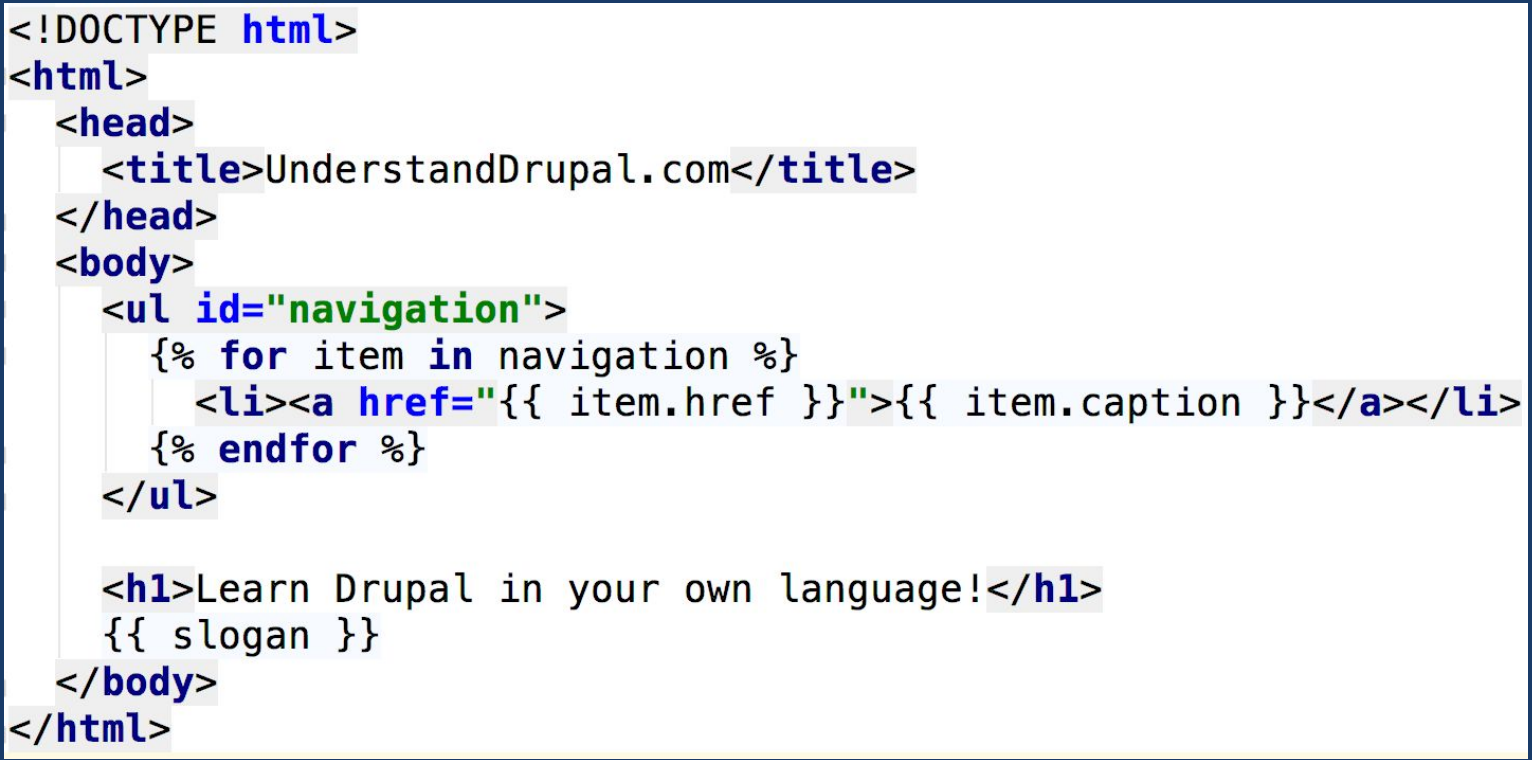
Author of *CSS Secrets*

Twig

Templates

They consists of HTML markup with **placeholders** that would be replaced by dynamic content.

CSS, JavaScript, and other assets are added as needed.



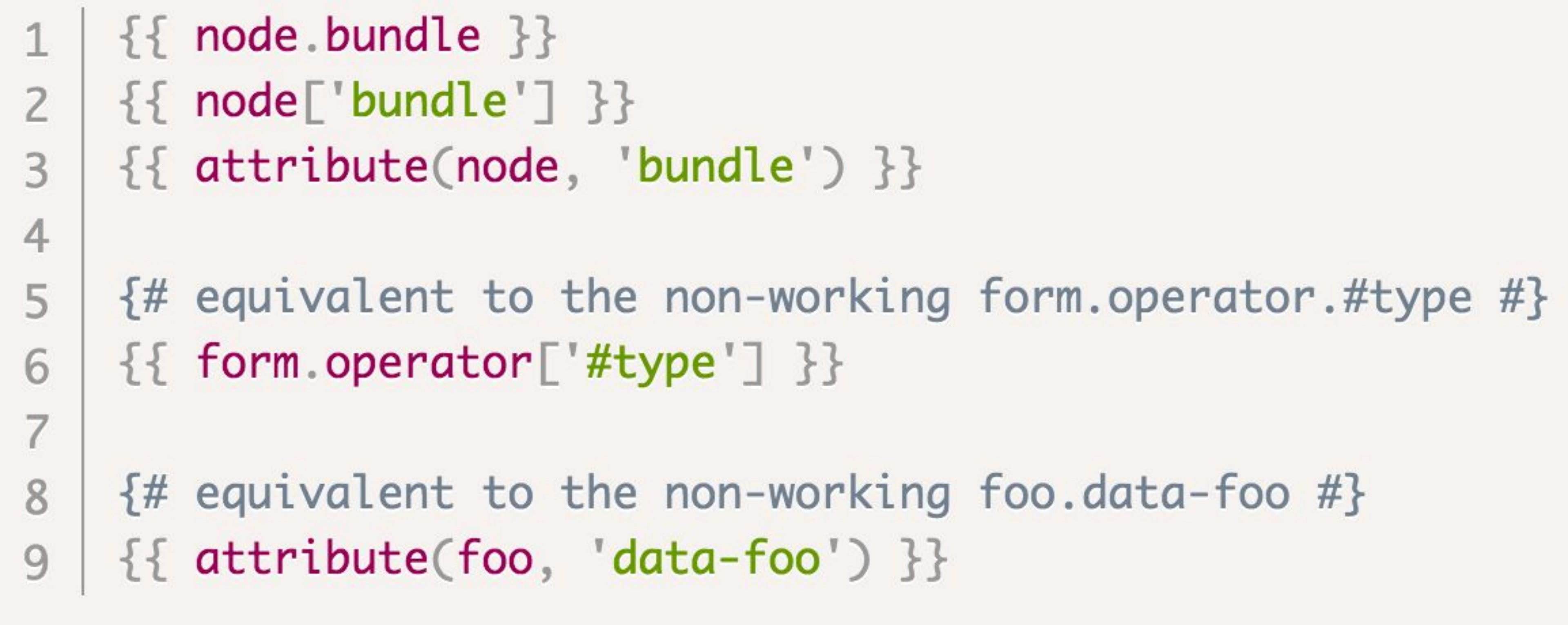
```
<!DOCTYPE html>
<html>
  <head>
    <title>UnderstandDrupal.com</title>
  </head>
  <body>
    <ul id="navigation">
      {% for item in navigation %}
        <li><a href="{{ item.href }}">{{ item.caption }}</a></li>
      {% endfor %}
    </ul>

    <h1>Learn Drupal in your own language!</h1>
    {{ slogan }}
  </body>
</html>
```

https://twig.sensiolabs.org/doc/templates.html

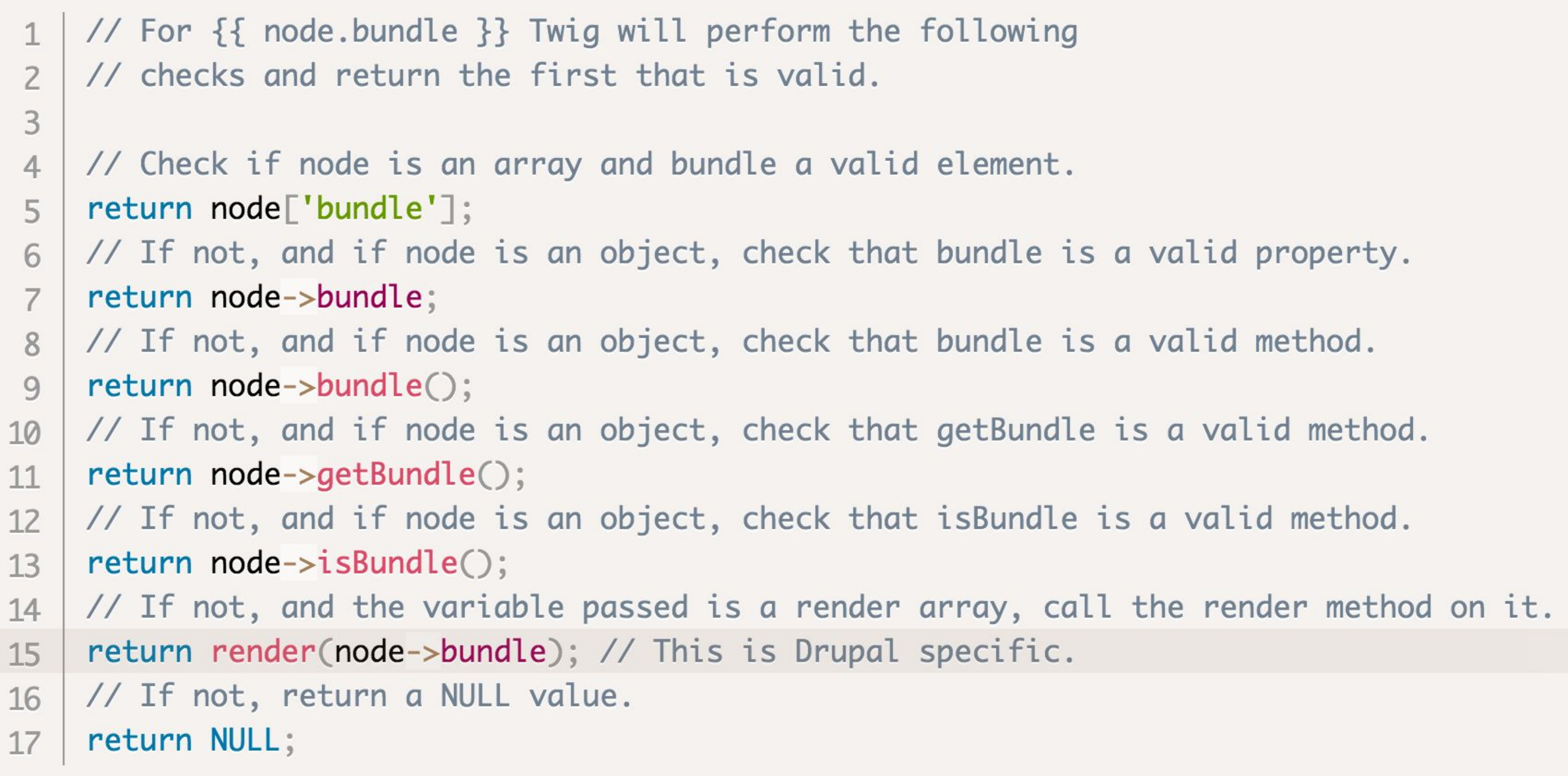
Twig Syntax

```
{{ print something }}

{%   do something  %}

{#    a comment    #}
```

Variables

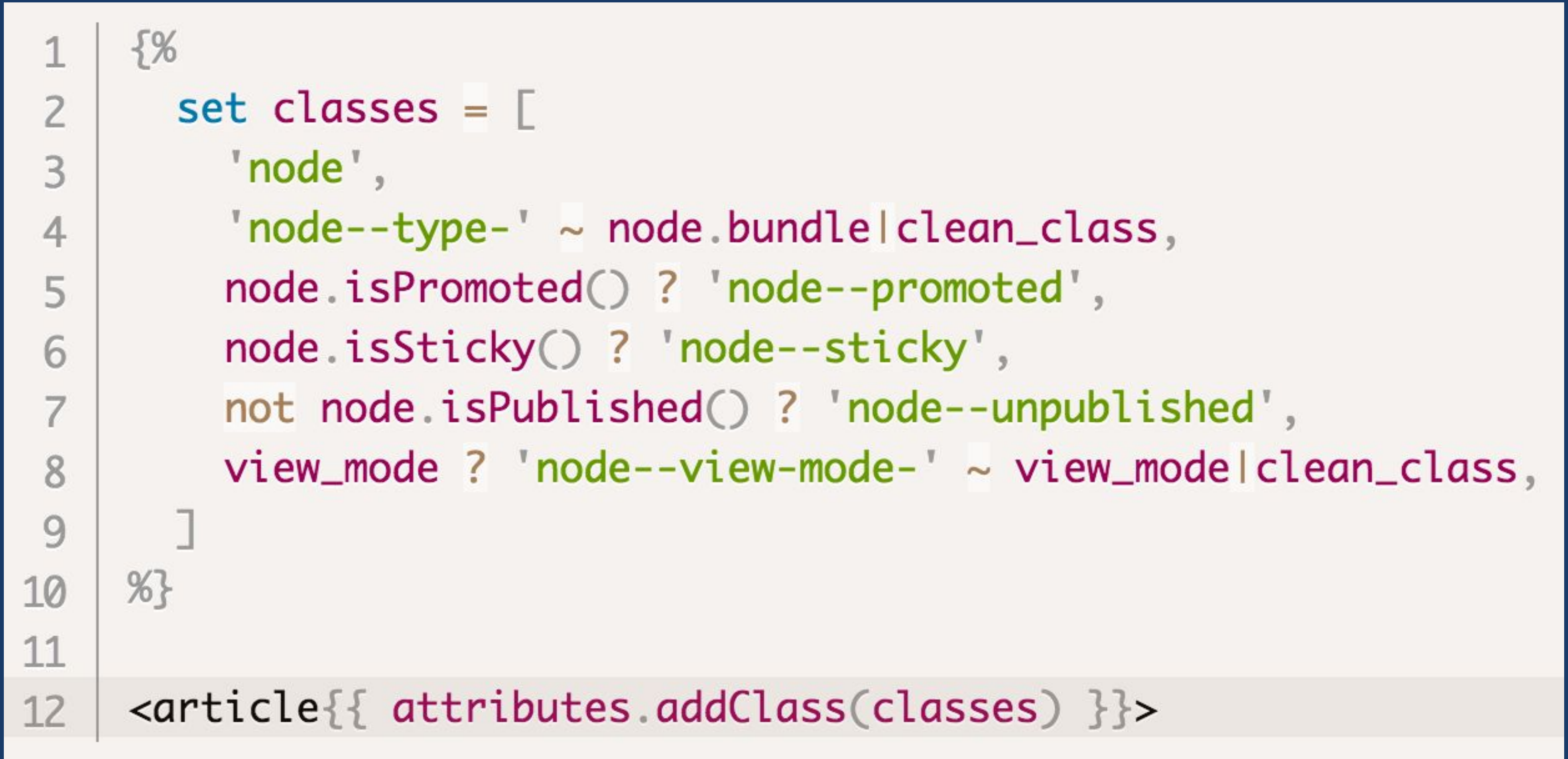


```
{{ node.bundle }}
{{ node['bundle'] }}
{{ attribute(node, 'bundle') }}

{# equivalent to the non-working form.operator.#type #}
{{ form.operator['#type'] }}

{# equivalent to the non-working foo.data-foo #}
{{ attribute(foo, 'data-foo') }}
```



```
// For {{ node.bundle }} Twig will perform the following
// checks and return the first that is valid.

// Check if node is an array and bundle a valid element.
return node['bundle'];
// If not, and if node is an object, check that bundle is a valid property.
return node->bundle;
// If not, and if node is an object, check that bundle is a valid method.
return node->bundle();
// If not, and if node is an object, check that getBundle is a valid method.
return node->getBundle();
// If not, and if node is an object, check that isBundle is a valid method.
return node->isBundle();
// If not, and the variable passed is a render array, call the render method on it.
return render(node->bundle); // This is Drupal specific.
// If not, return a NULL value.
return NULL;
```

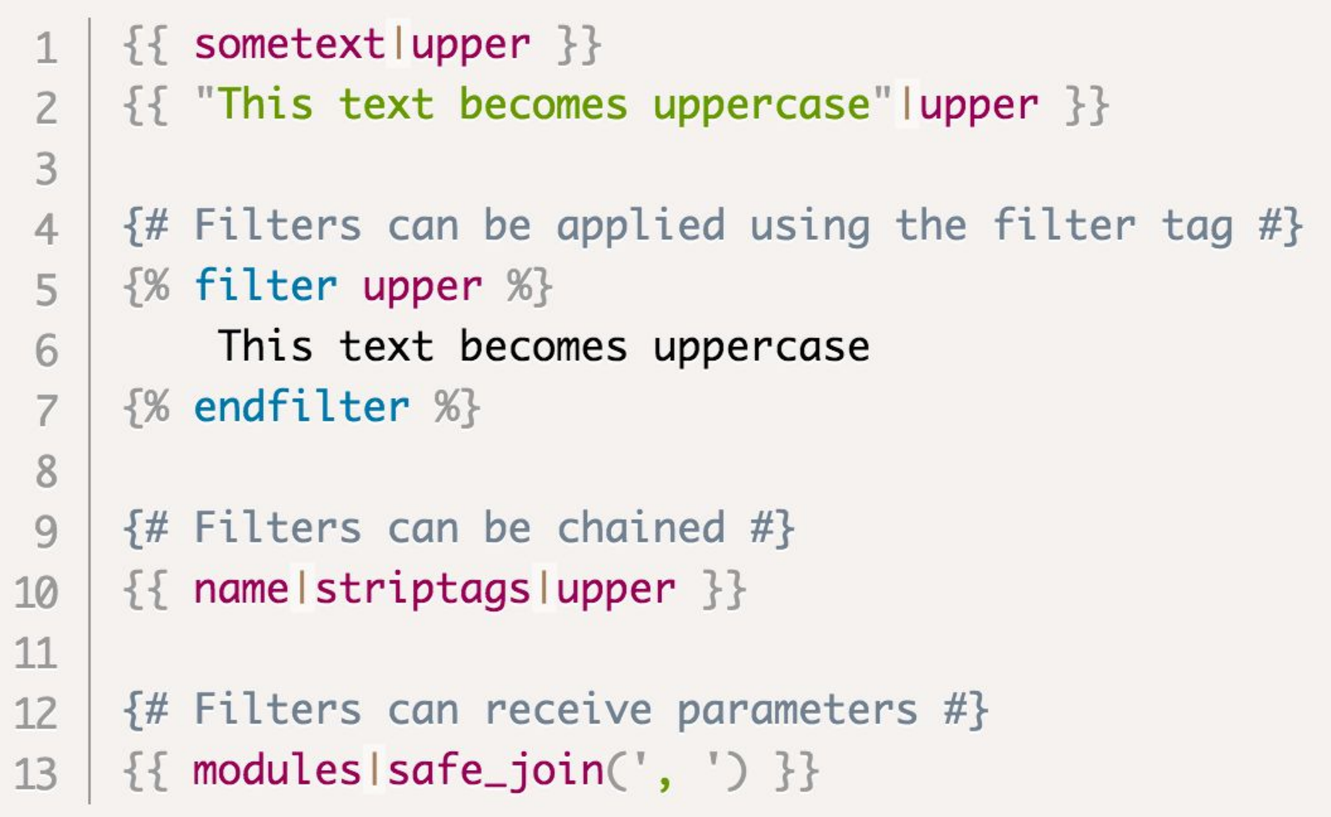
Setting Variables



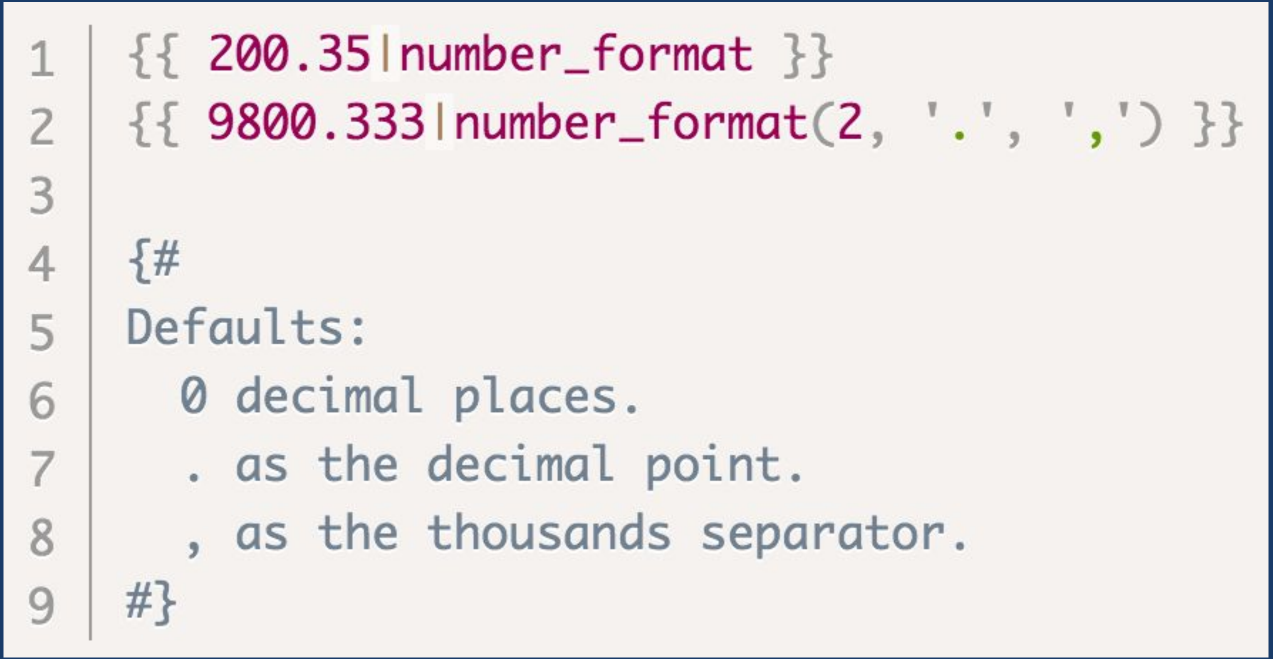
```
{%
  set classes = [
    'node',
    'node--type-' ~ node.bundle|clean_class,
    node.isPromoted() ? 'node--promoted',
    node.isSticky() ? 'node--sticky',
    not node.isPublished() ? 'node--unpublished',
    view_mode ? 'node--view-mode-' ~ view_mode|clean_class,
  ]
%}

<article{{ attributes.addClass(classes) }}>
```

Filters

Drupal specific http://bit.ly/2kcWV0K



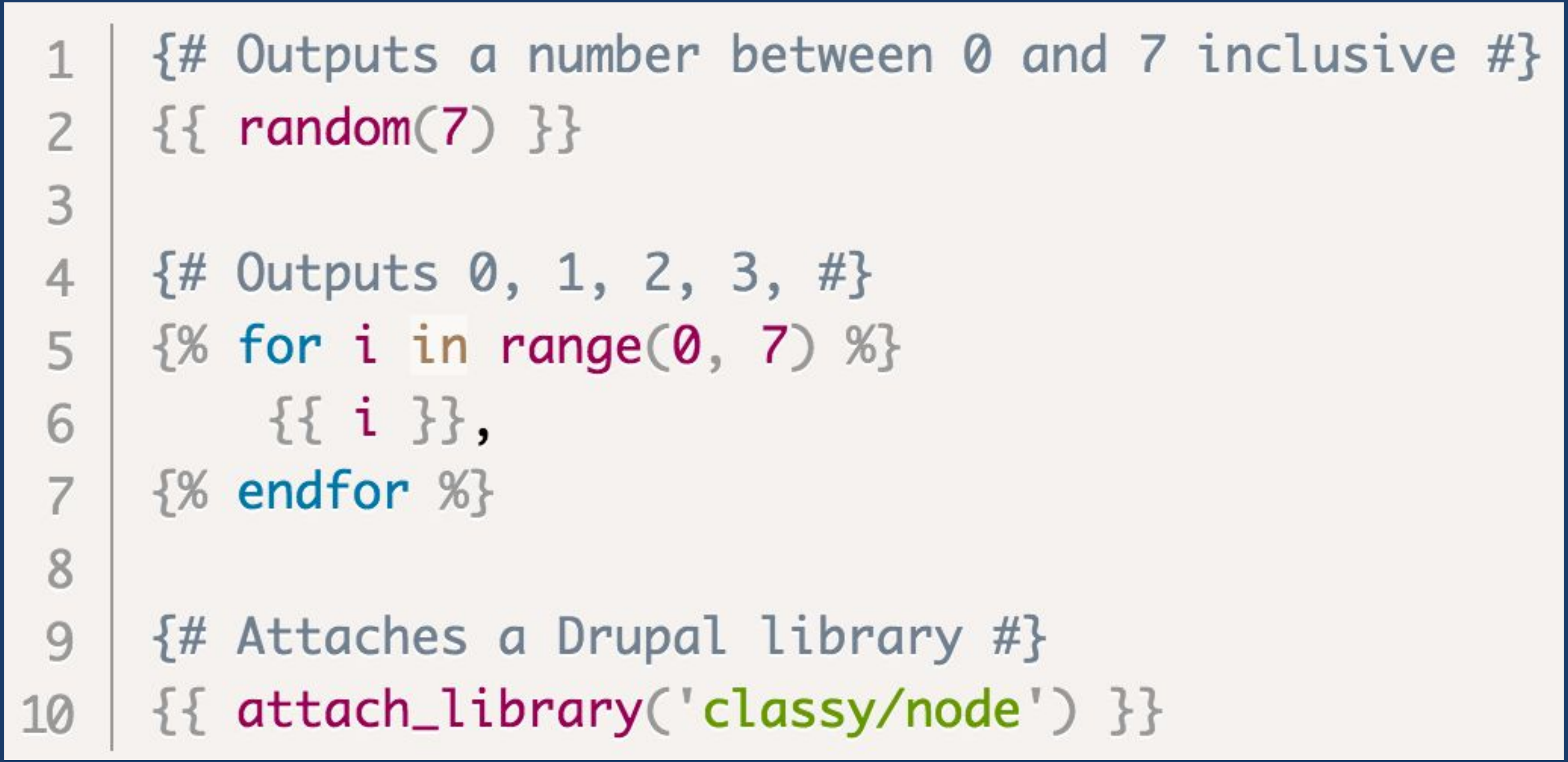
```
{{ sometext|upper }}
{{ "This text becomes uppercase"|upper }}

{# Filters can be applied using the filter tag #}
{% filter upper %}
    This text becomes uppercase
{% endfilter %}

{# Filters can be chained #}
{{ name|striptags|upper }}

{# Filters can receive parameters #}
{{ modules|safe_join(', ') }}
```



```
{{ 200.35|number_format }}
{{ 9800.333|number_format(2, '.', ',') }}

{#
Defaults:
  0 decimal places.
  . as the decimal point.
  , as the thousands separator.
#}
```

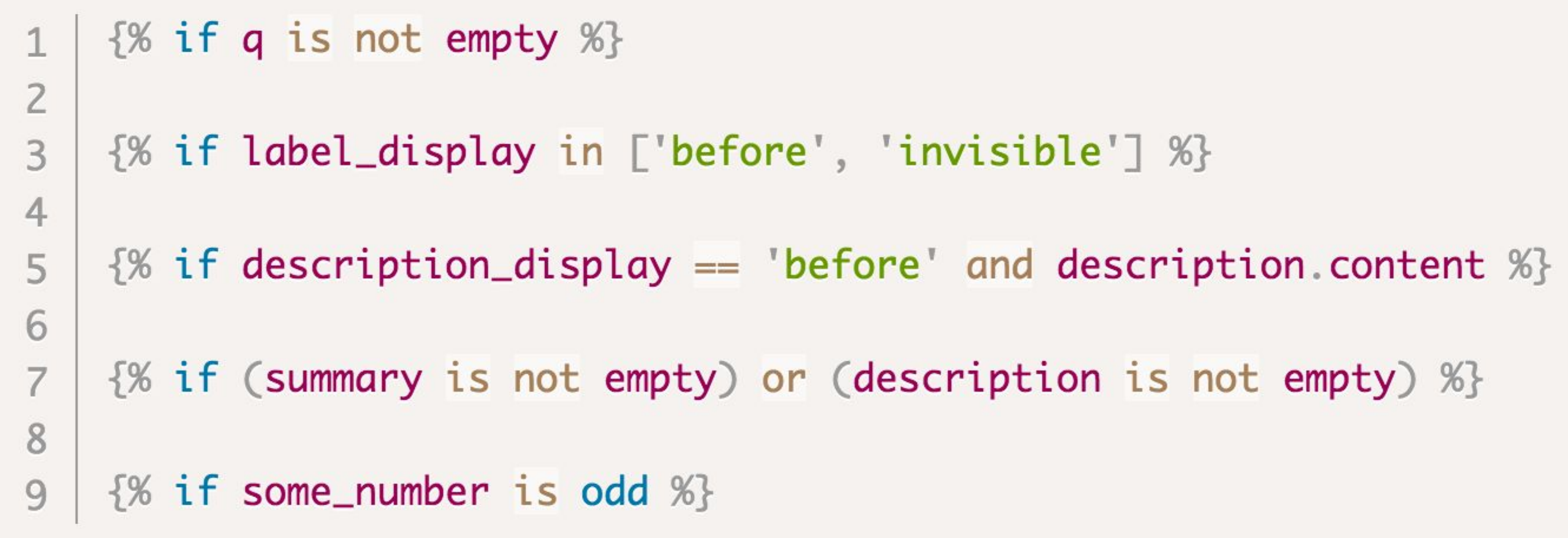
Functions

Drupal specific http://bit.ly/2jLkizT



```
{# Outputs a number between 0 and 7 inclusive #}
{{ random(7) }}

{# Outputs 0, 1, 2, 3, #}
{% for i in range(0, 7) %}
    {{ i }},
{% endfor %}

{# Attaches a Drupal library #}
{{ attach_library('classy/node') }}
```

Tests

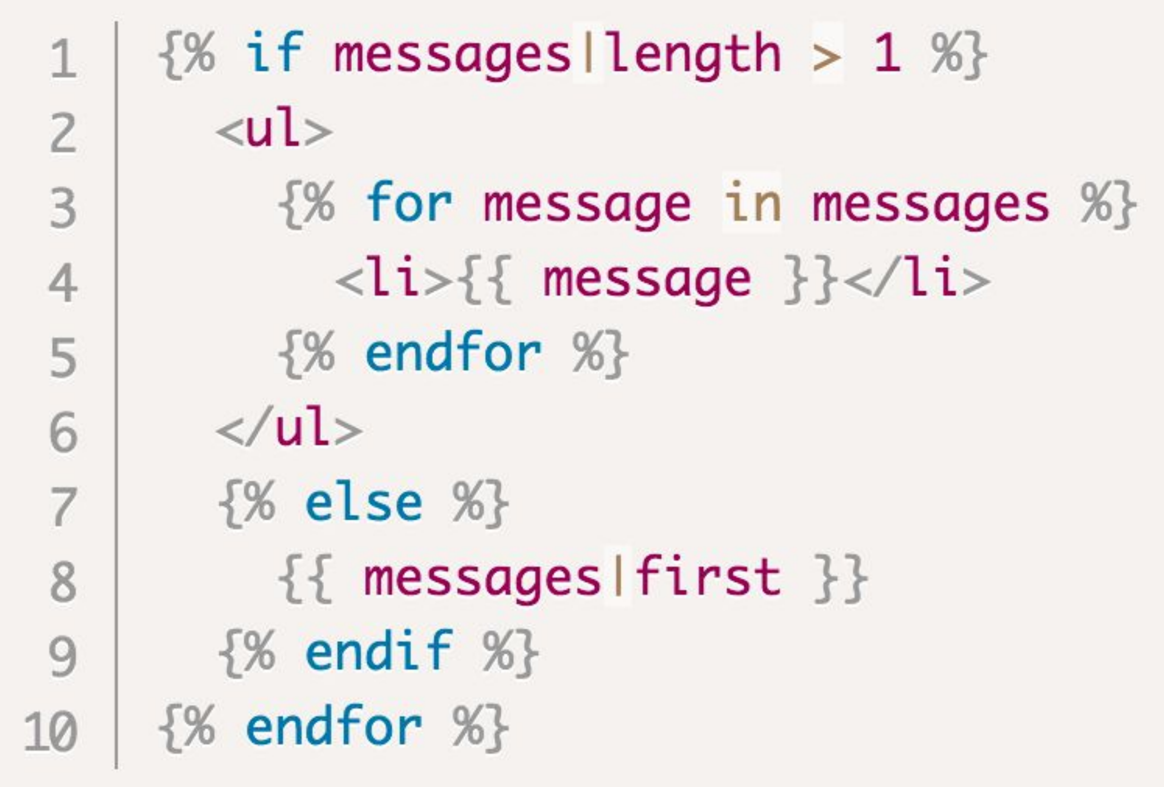


```
{% if q is not empty %}

{% if label_display in ['before', 'invisible'] %}

{% if description_display == 'before' and description.content %}

{% if (summary is not empty) or (description is not empty) %}

{% if some_number is odd %}
```

Filter or Function or Test?

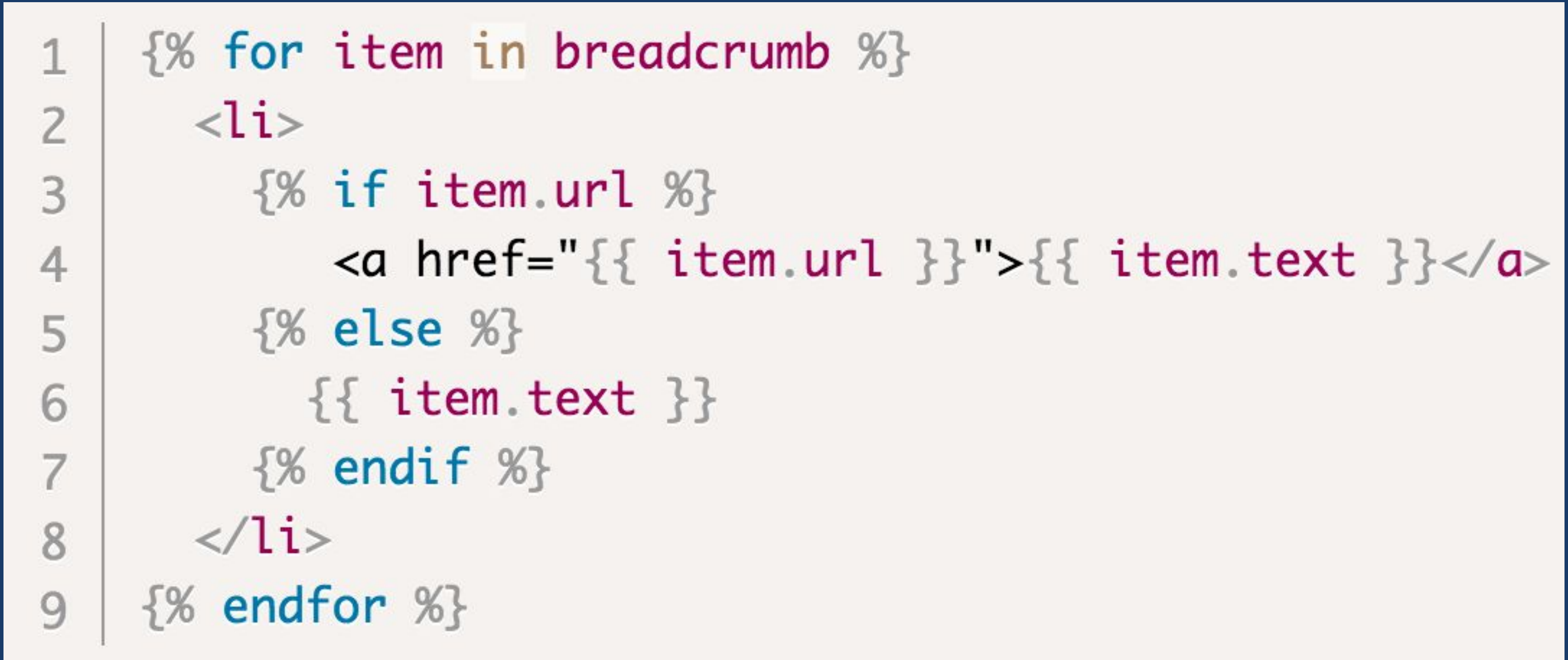
	Filter	Function	Test
Requires an expression preceding it?	Yes	No	Yes
Can receive arguments?	Yes	Yes	Yes
Return value	Scalar	Mixed	Boolean

Control Structures



```
{% if messages|length > 1 %}
  <ul>
    {% for message in messages %}
      <li>{{ message }}</li>
    {% endfor %}
  </ul>
  {% else %}
    {{ messages|first }}
  {% endif %}
{% endfor %}
```

Control Structures

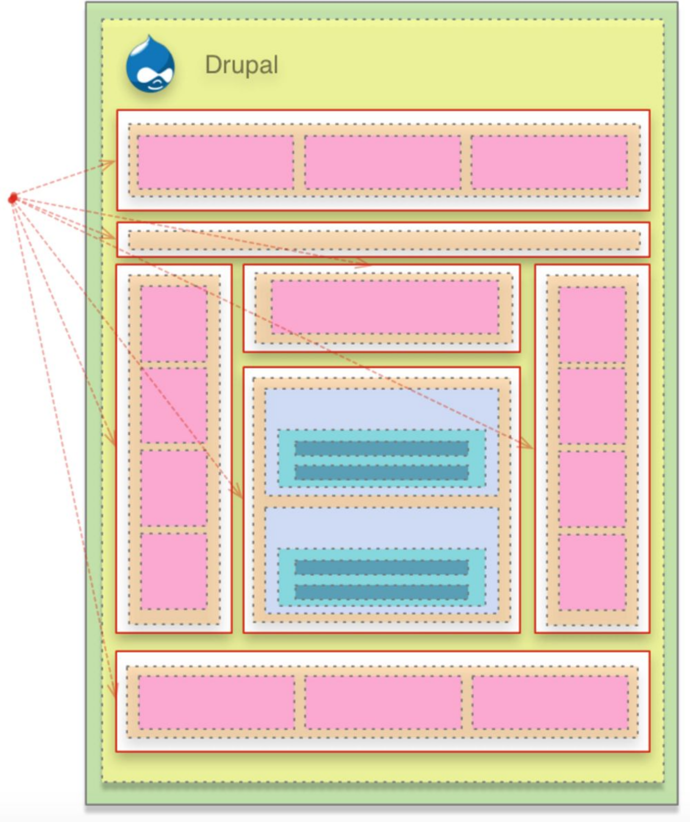


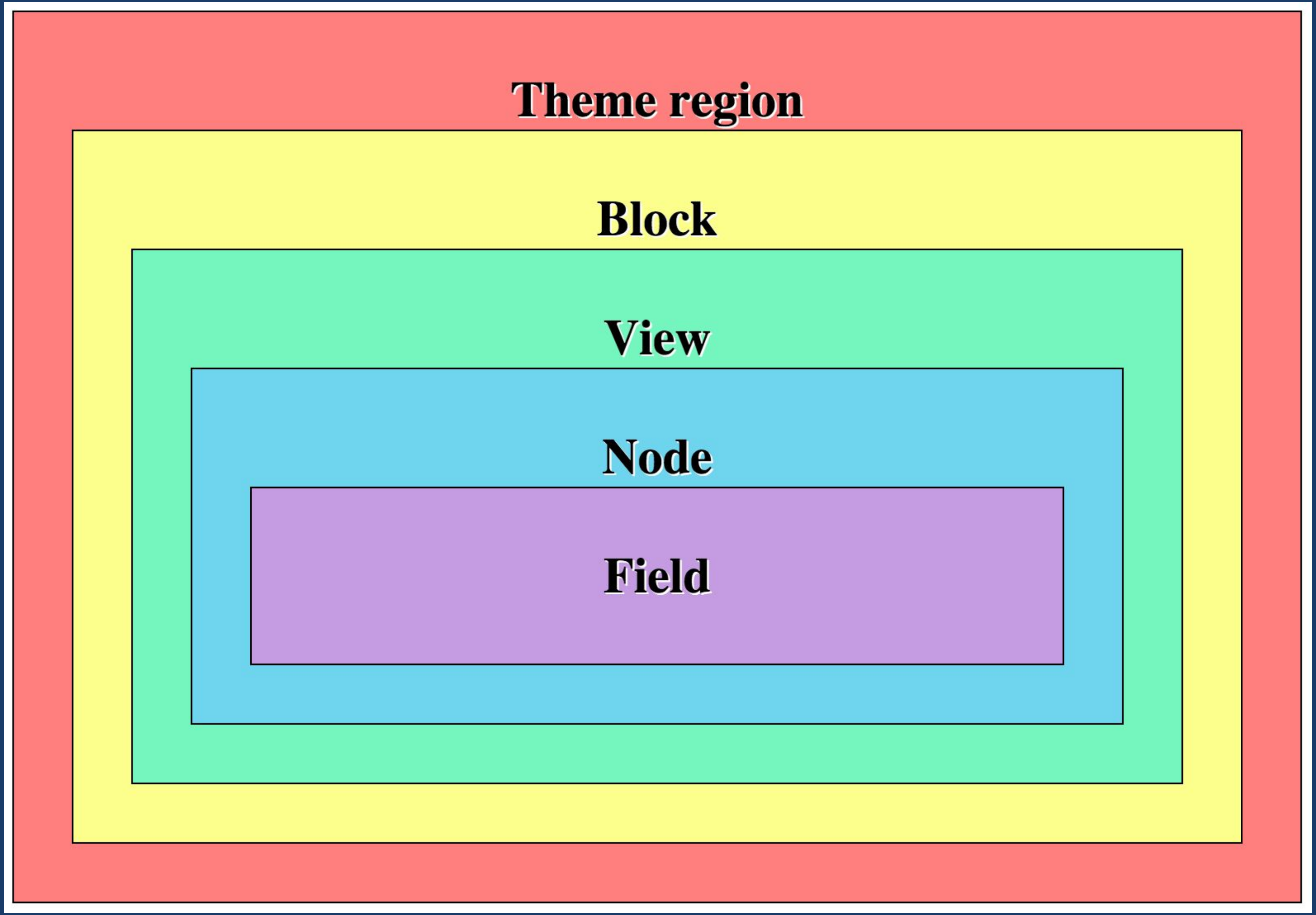
```
{% for item in breadcrumb %}
  <li>
    {% if item.url %}
       <a href="{{ item.url }}">{{ item.text }}</a>
    {% else %}
      {{ item.text }}
    {% endif %}
  </li>
{% endfor %}
```

Drupal Templates

Drupal templates are like onions

Drupal templates come in layers





Theme region

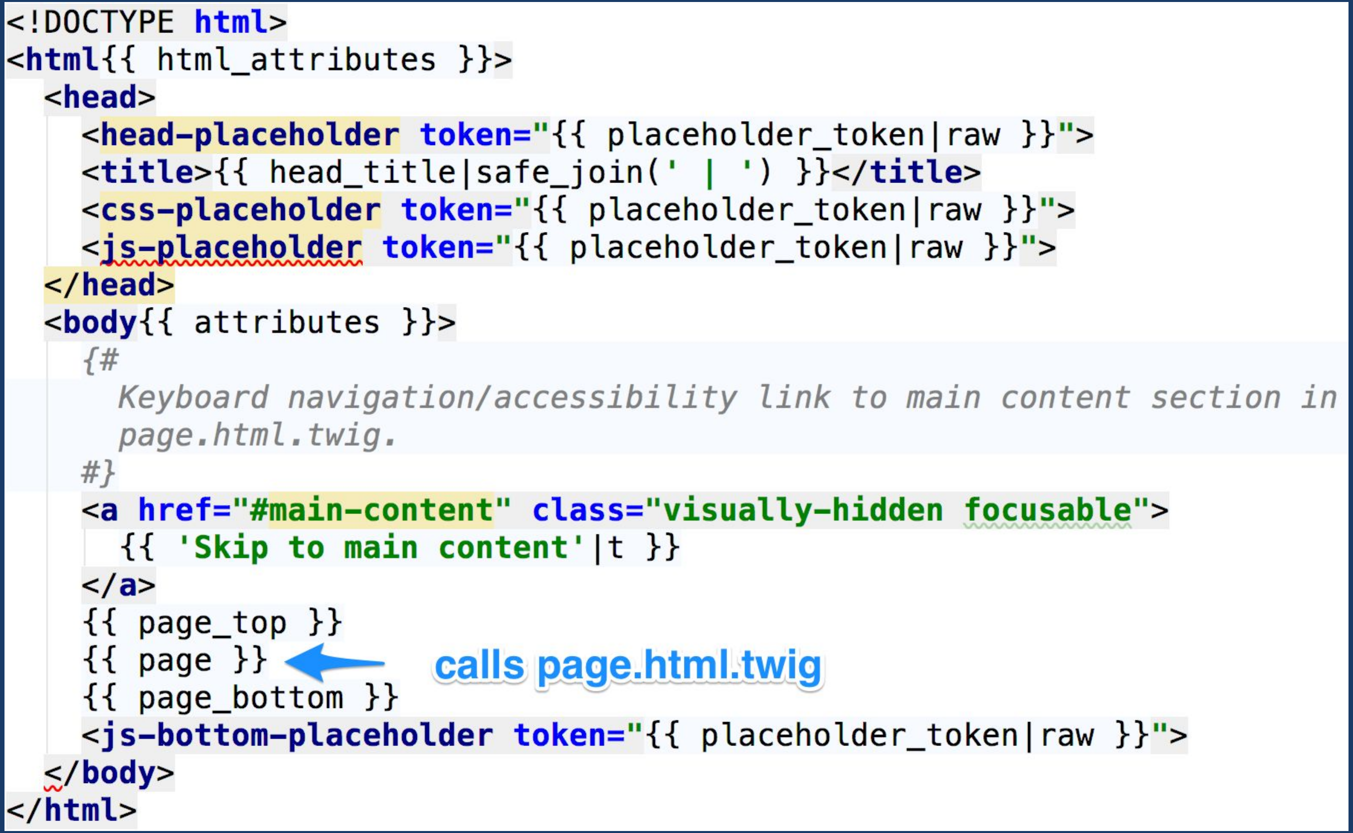
Block

View

Node

Field

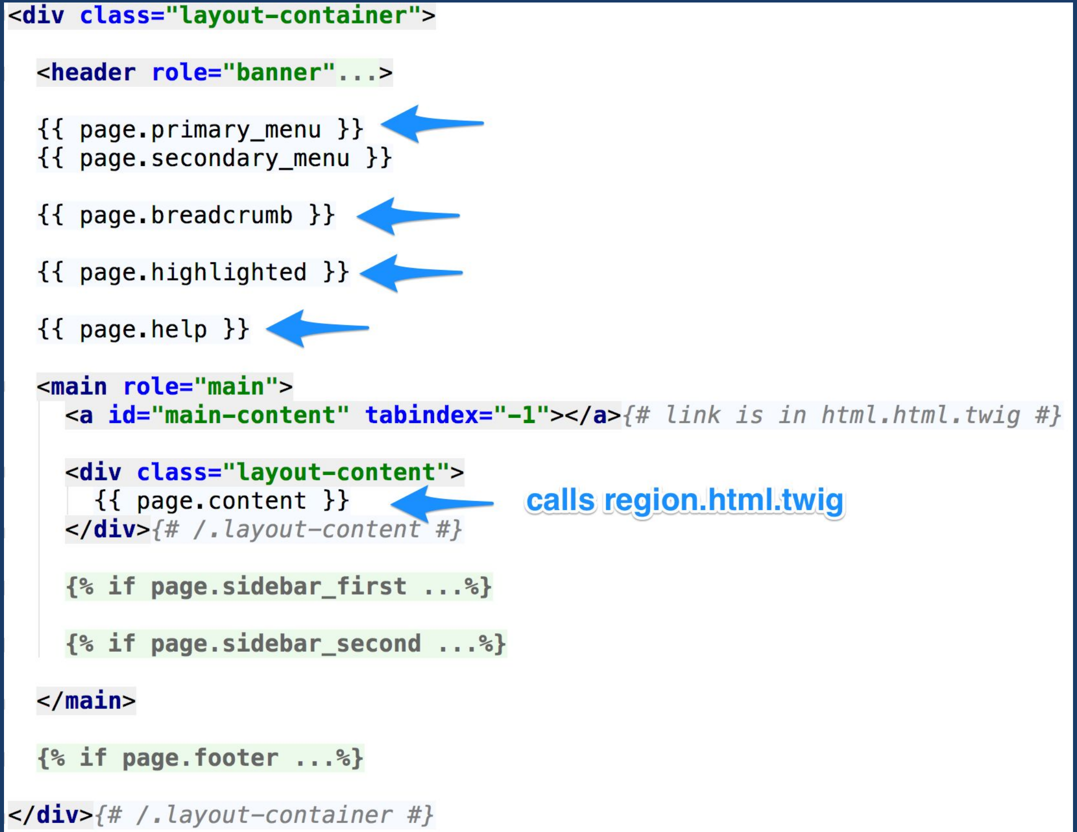
html.html.twig



```
<!DOCTYPE html>
<html{{ html_attributes }}>
  <head>
    <head-placeholder token="{{ placeholder_token|raw }}">
    <title>{{ head_title|safe_join(' | ') }}</title>
    <css-placeholder token="{{ placeholder_token|raw }}">
    <js-placeholder token="{{ placeholder_token|raw }}">
  </head>
  <body{{ attributes }}>
    {#
      Keyboard navigation/accessibility link to main content section in
      page.html.twig.
    #}
    <a href="#main-content" class="visually-hidden focusable">
      {{ 'Skip to main content'|t }}
    </a>
    {{ page_top }}
    {{ page }}
    {{ page_bottom }}
    <js-bottom-placeholder token="{{ placeholder_token|raw }}">
  </body>
</html>
```

calls page.html.twig

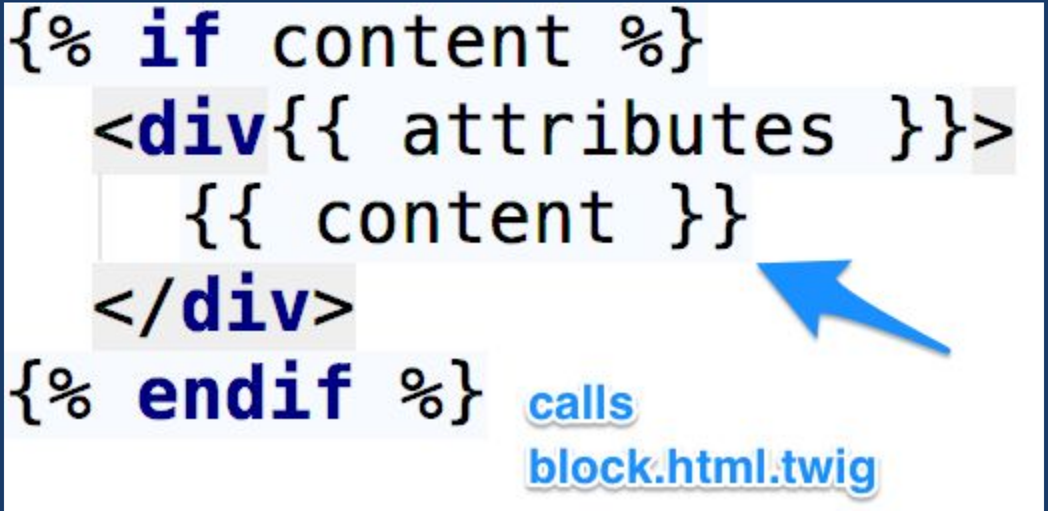
page.html.twig



```
<div class="layout-container">

  <header role="banner"...>

  {{ page.primary_menu }}
  {{ page.secondary_menu }}

  {{ page.breadcrumb }}

  {{ page.highlighted }}

  {{ page.help }}

  <main role="main">
    <a id="main-content" tabindex="-1"></a>{# link is in html.html.twig #}

    <div class="layout-content">
      {{ page.content }}
    </div>{# /.layout-content #}

    {% if page.sidebar_first ...%}

    {% if page.sidebar_second ...%}

  </main>

  {% if page.footer ...%}

</div>{# /.layout-container #}
```

calls region.html.twig

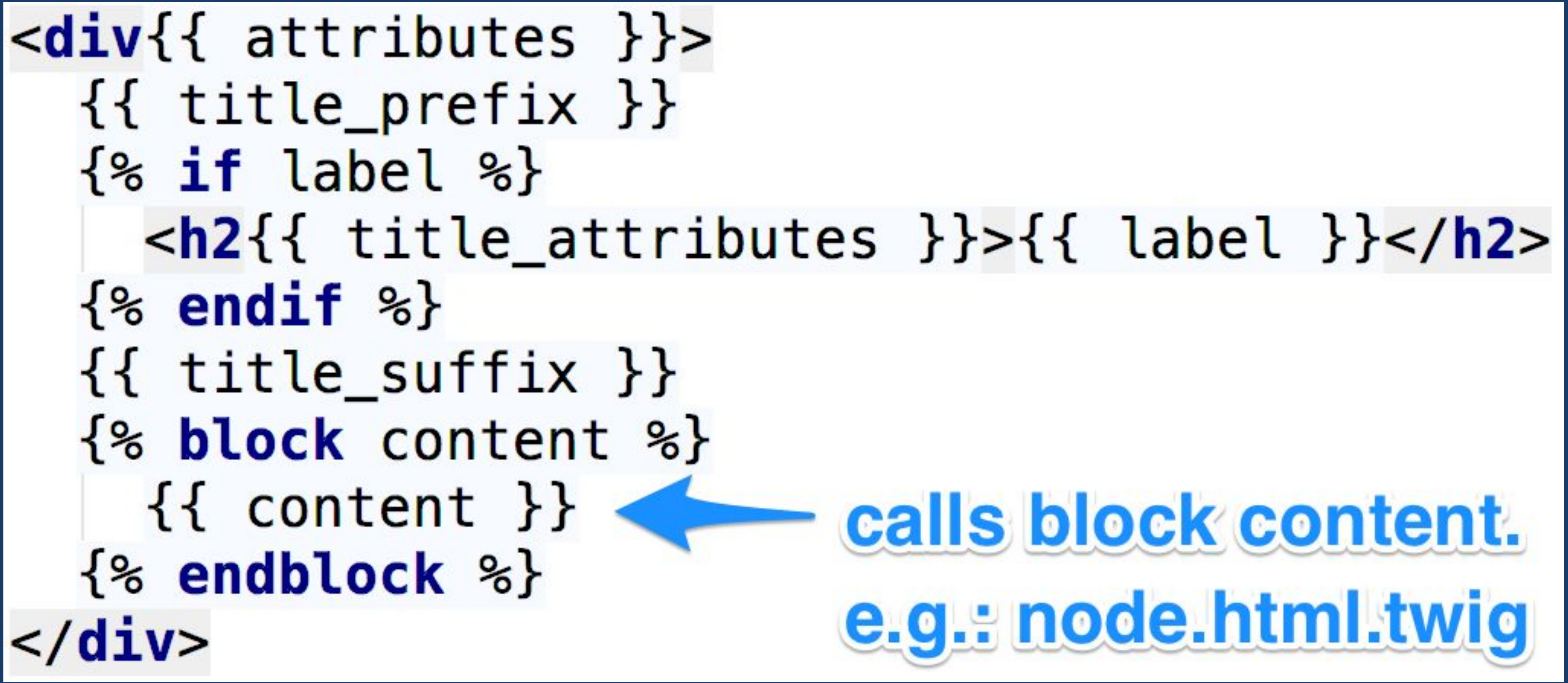
region.html.twig



```
{% if content %}
  <div{{ attributes }}>
    {{ content }}
  </div>
{% endif %}
```

calls
block.html.twig

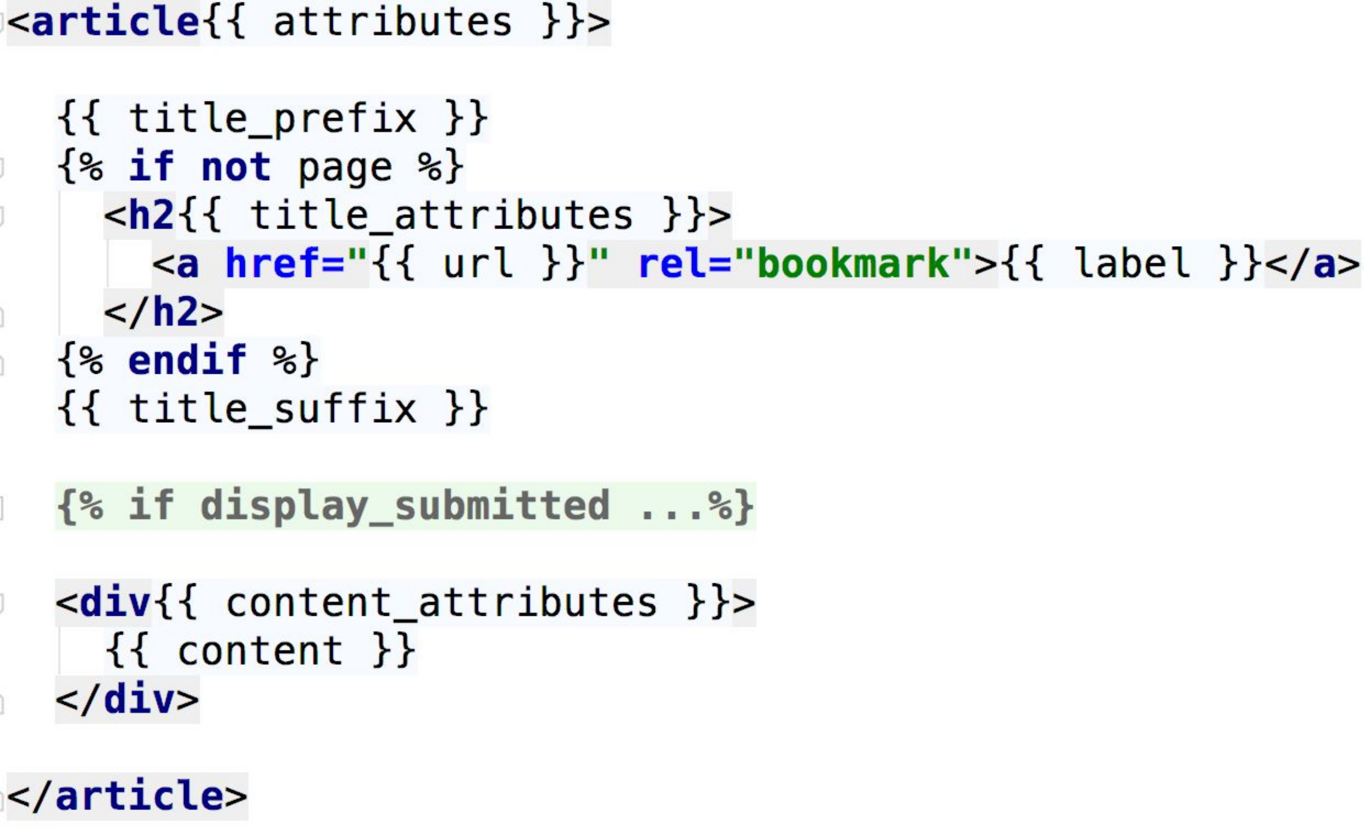
block.html.twig



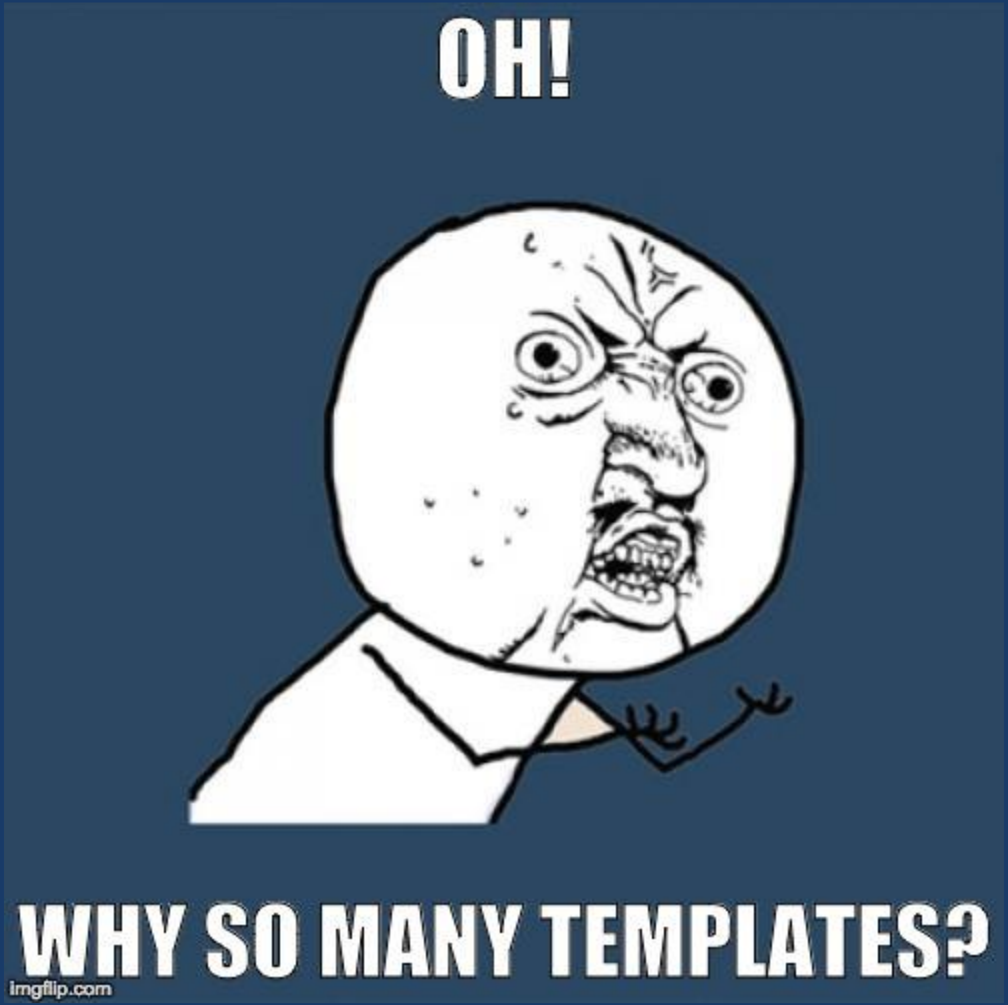
```
<div{{ attributes }}>
  {{ title_prefix }}
  {% if label %}
    <h2{{ title_attributes }}>{{ label }}</h2>
  {% endif %}
  {{ title_suffix }}
  {% block content %}
    {{ content }}
  {% endblock %}
</div>
```

calls block content.
e.g.: node.html.twig

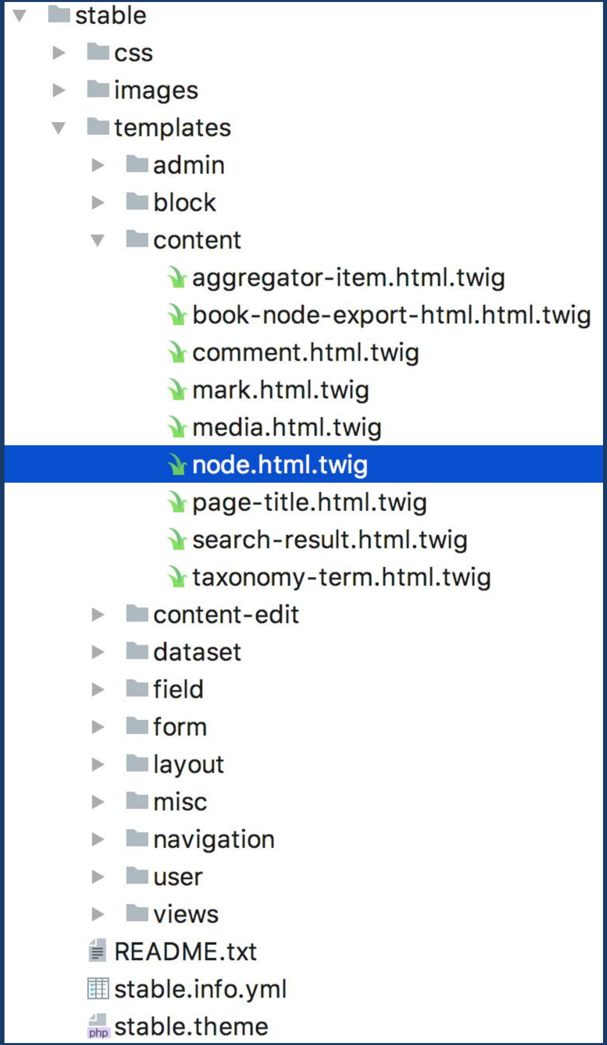
node.html.twig



```
<article{{ attributes }}>

  {{ title_prefix }}
  {% if not page %}
    <h2{{ title_attributes }}>
      <a href="{{ url }}" rel="bookmark">{{ label }}</a>
    </h2>
  {% endif %}
  {{ title_suffix }}

  {% if display_submitted ...%}

  <div{{ content_attributes }}>
    {{ content }}
  </div>

</article>
```


OH!
WHY SO MANY TEMPLATES?
imgflip.com

agaric



```
▼ stable
    ▶ css
    ▶ images
    ▼ templates
        ▶ admin
        ▶ block
        ▼ content
            aggregator-item.html.twig
            book-node-export-html.html.twig
            comment.html.twig
            mark.html.twig
            media.html.twig
            node.html.twig
            page-title.html.twig
            search-result.html.twig
            taxonomy-term.html.twig
        ▶ content-edit
        ▶ dataset
        ▶ field
        ▶ form
        ▶ layout
        ▶ misc
        ▶ navigation
        ▶ user
        ▶ views
    README.txt
    stable.info.yml
    stable.theme
```

166 templates in the **stable** Core theme

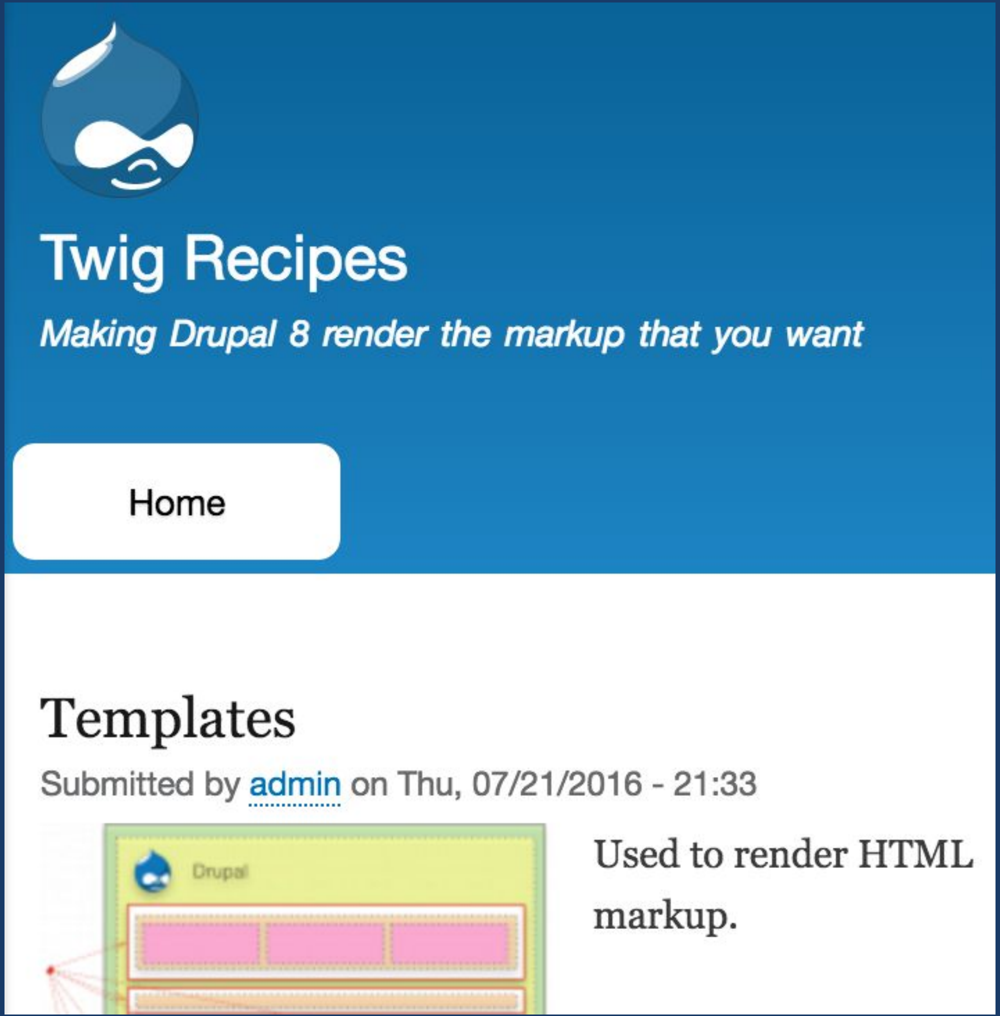
Overriding Drupal's Markup

Exercise

Change the tagline date format to

[month] [day], [year]

Example: July 21, 2016



Twig Recipes

Making Drupal 8 render the markup that you want

Templates

Submitted by admin on Thu, 07/21/2016 - 21:33

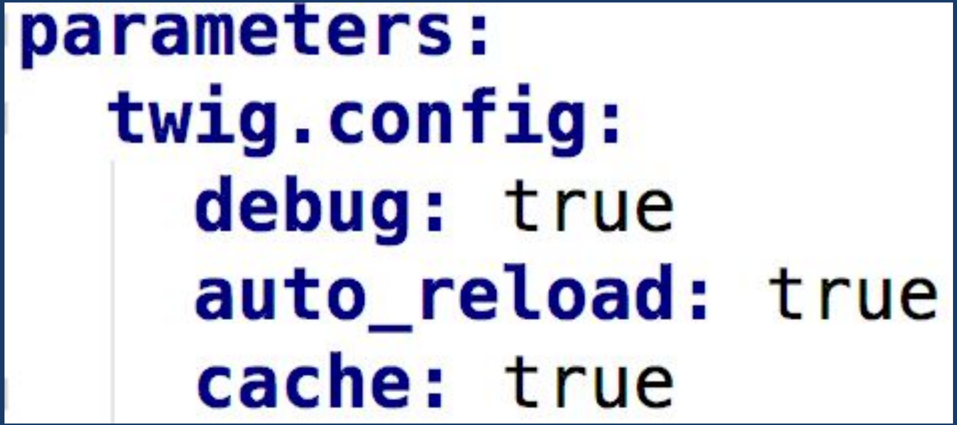
Used to render HTML markup.

Steps to Override Default Markup

- Enable theme debug.
- Locate the proper template to override.
- Copy the template to your theme.
- Use a template file name suggestion if needed.
- Make your changes.
- Clear caches.
- Rinse and repeat.
- Disable theme debug.

Theme Debug

sites/default/services.yml

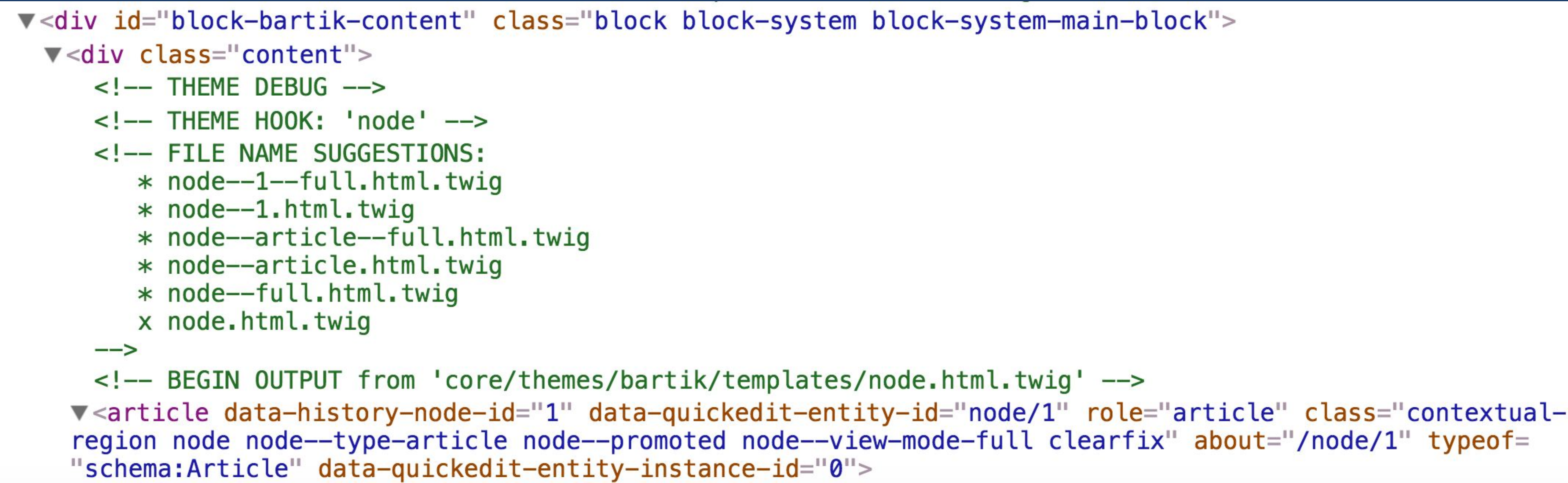


```
parameters:
  twig.config:
    debug: true
    auto_reload: true
    cache: true
```

https://www.drupal.org/node/1906392

https://www.drupal.org/node/2358785

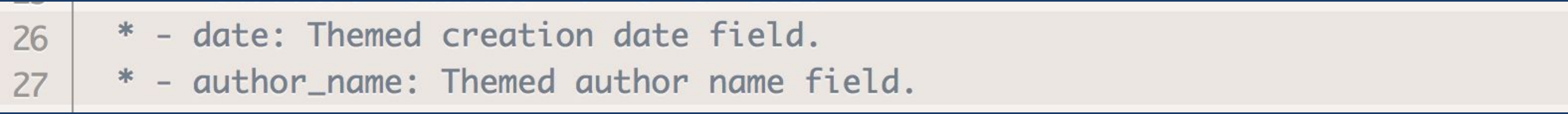
Template Suggestions



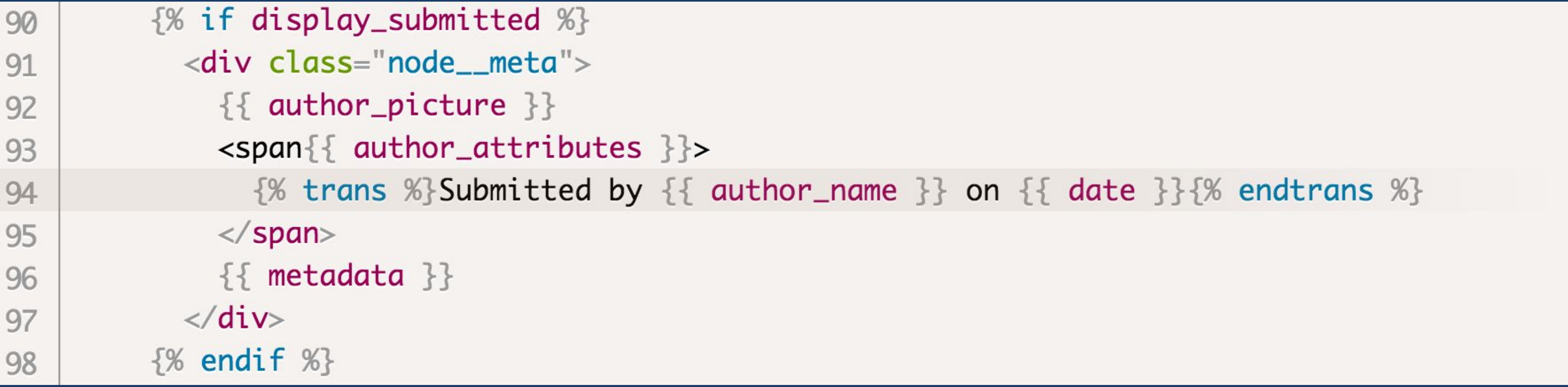
```
▼<div id="block-bartik-content" class="block block-system block-system-main-block">
  ▼<div class="content">
      <!-- THEME DEBUG -->
      <!-- THEME HOOK: 'node' -->
      <!-- FILE NAME SUGGESTIONS:
         * node--1--full.html.twig
         * node--1.html.twig
         * node--article--full.html.twig
         * node--article.html.twig
         * node--full.html.twig
         x node.html.twig
      -->
      <!-- BEGIN OUTPUT from 'core/themes/bartik/templates/node.html.twig' -->
    ▼<article data-history-node-id="1" data-quickedit-entity-id="node/1" role="article" class="contextual-
    region node node--type-article node--promoted node--view-mode-full clearfix" about="/node/1" typeof=
    "schema:Article" data-quickedit-entity-instance-id="0">
```

https://www.drupal.org/node/2354645

https://bit.ly/2KgLp3n (views templates suggestions)

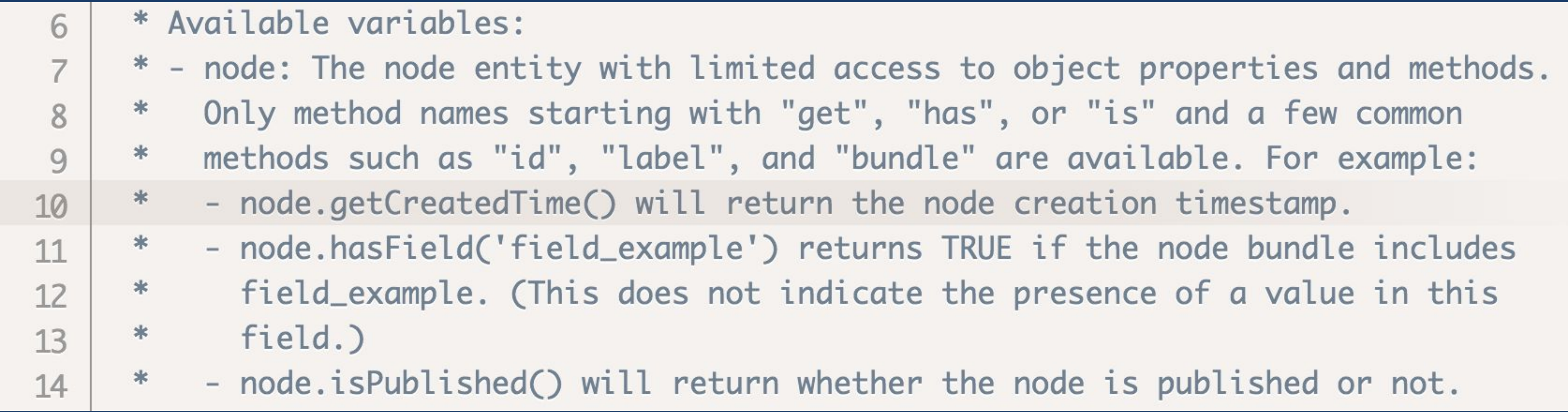


```
 * - date: Themed creation date field.
 * - author_name: Themed author name field.
```



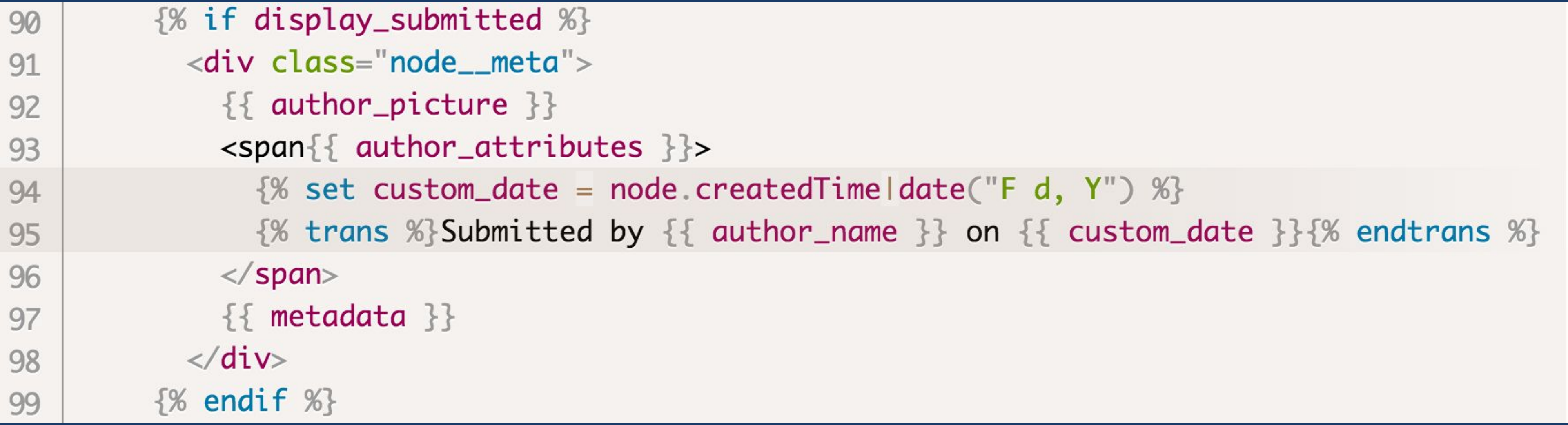
```
    {% if display_submitted %}
      <div class="node__meta">
        {{ author_picture }}
        <span{{ author_attributes }}>
          {% trans %}Submitted by {{ author_name }} on {{ date }}{% endtrans %}
        </span>
        {{ metadata }}
      </div>
    {% endif %}
```

./core/themes/bartik/templates/node.html.twig



```
 * Available variables:
 * - node: The node entity with limited access to object properties and methods.
 *   Only method names starting with "get", "has", or "is" and a few common
 *   methods such as "id", "label", and "bundle" are available. For example:
 *   - node.getCreatedTime() will return the node creation timestamp.
 *   - node.hasField('field_example') returns TRUE if the node bundle includes
 *     field_example. (This does not indicate the presence of a value in this
 *     field.)
 *   - node.isPublished() will return whether the node is published or not.
```

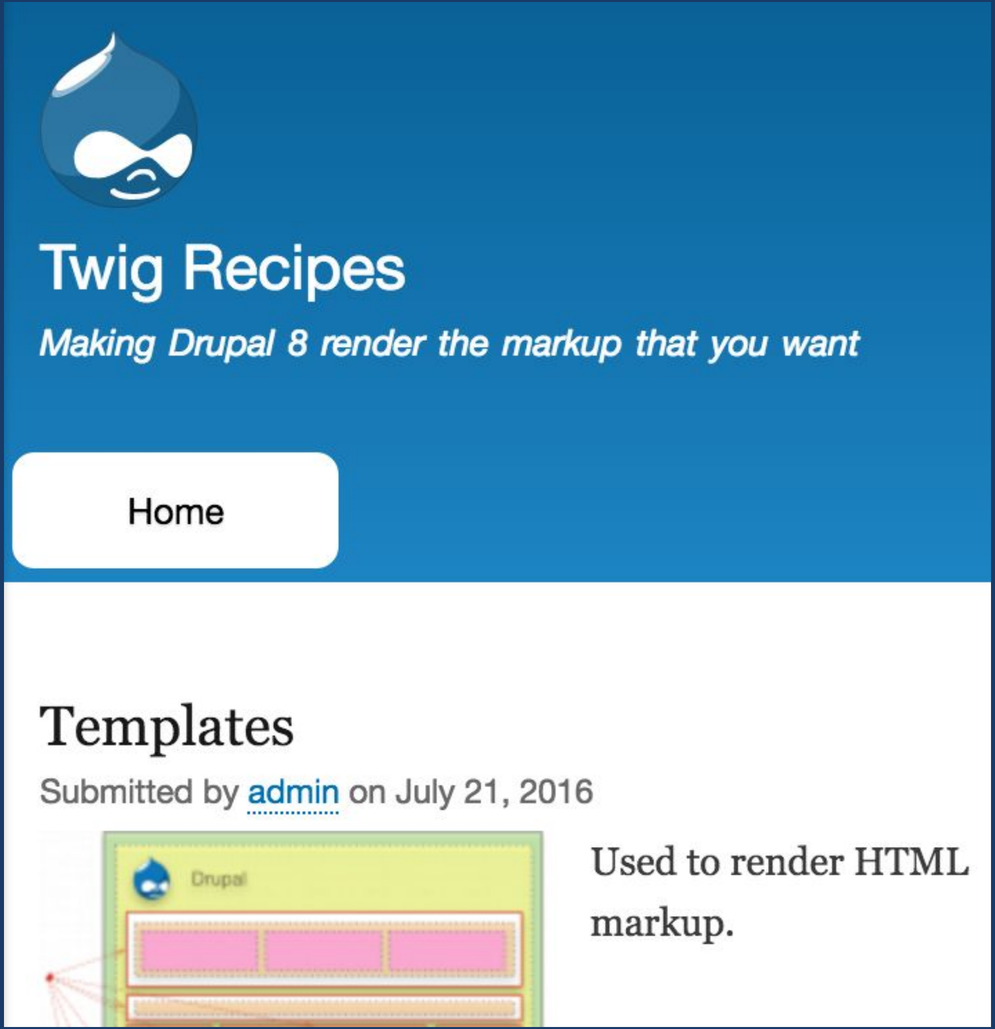
./core/themes/bartik/templates/node.html.twig



```
{% if display_submitted %}
  <div class="node__meta">
    {{ author_picture }}
    <span{{ author_attributes }}>
      {% set custom_date = node.createdTime|date("F d, Y") %}
      {% trans %}Submitted by {{ author_name }} on {{ custom_date }}{% endtrans %}
    </span>
    {{ metadata }}
  </div>
{% endif %}
```

./themes/YOUR_THEME_NAME/templates/node.html.twig

+ Rebuild caches



Twig Recipes

Making Drupal 8 render the markup that you want

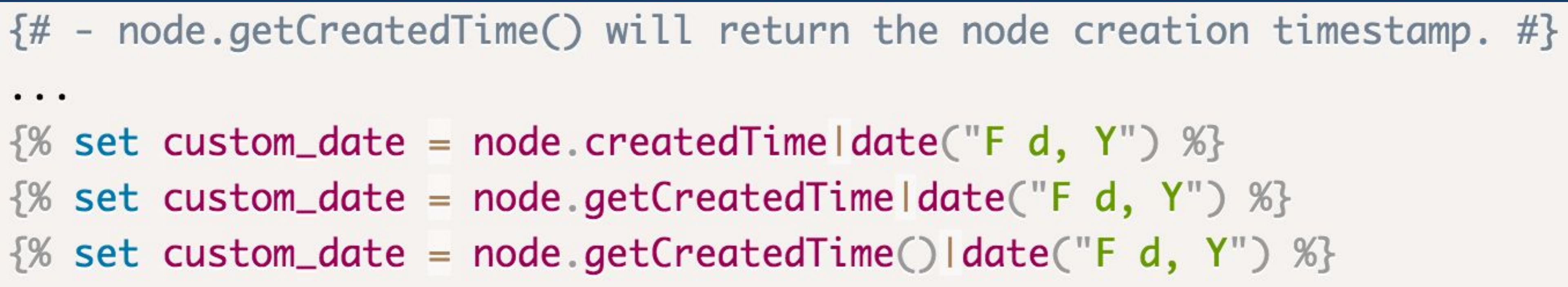
Home

Templates

Submitted by admin on July 21, 2016

Used to render HTML markup.

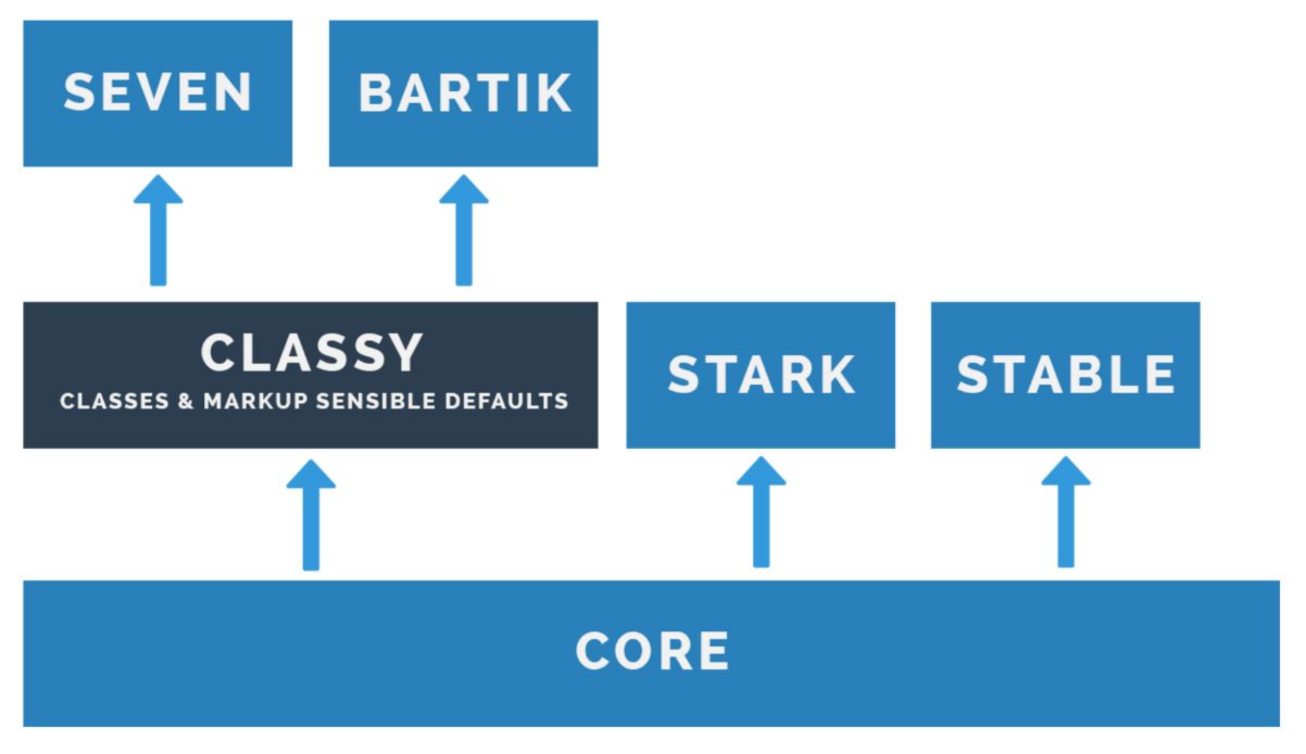
All of these are equivalent



```
{# - node.getCreatedTime() will return the node creation timestamp. #}
...
{% set custom_date = node.createdTime|date("F d, Y") %}
{% set custom_date = node.getCreatedTime|date("F d, Y") %}
{% set custom_date = node.getCreatedTime()|date("F d, Y") %}
```

Warning!

For demonstration purposes only. DO NOT subtheme Bartik.



Twig Recipes

Content vs Node Variables

- The **content** variable has render arrays for fields as defined in the “Manage display” section of a content type. These are the view mode settings.
- The **node** variable is the node object. It contains the fields values and referenced entities as stored in the database.

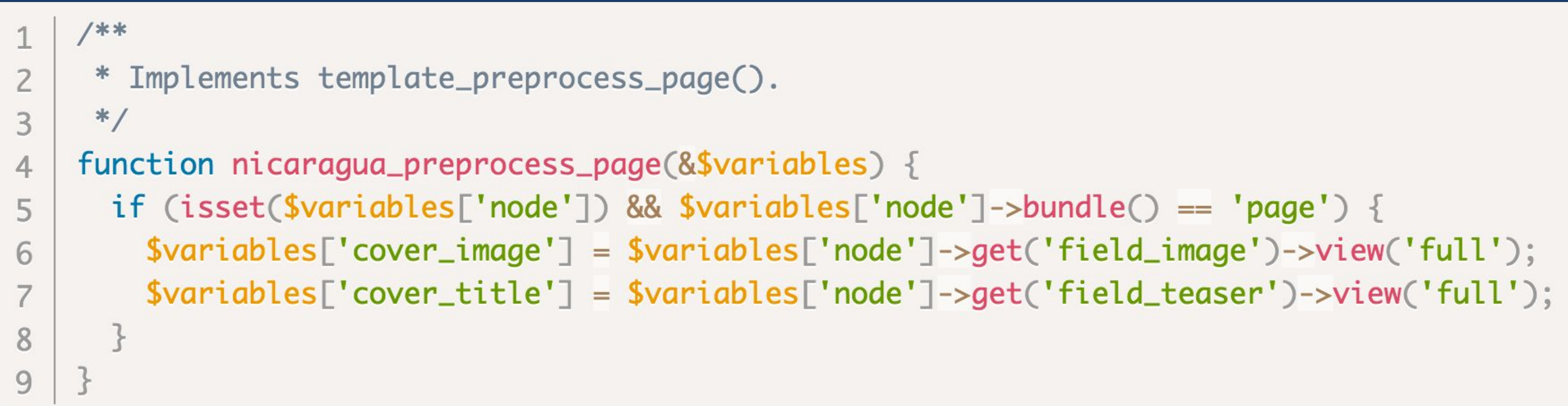
http://bit.ly/2sMUwg8

By Sascha Grossenbacher @berdir

1. How can I pass information from Drupal to Twig?

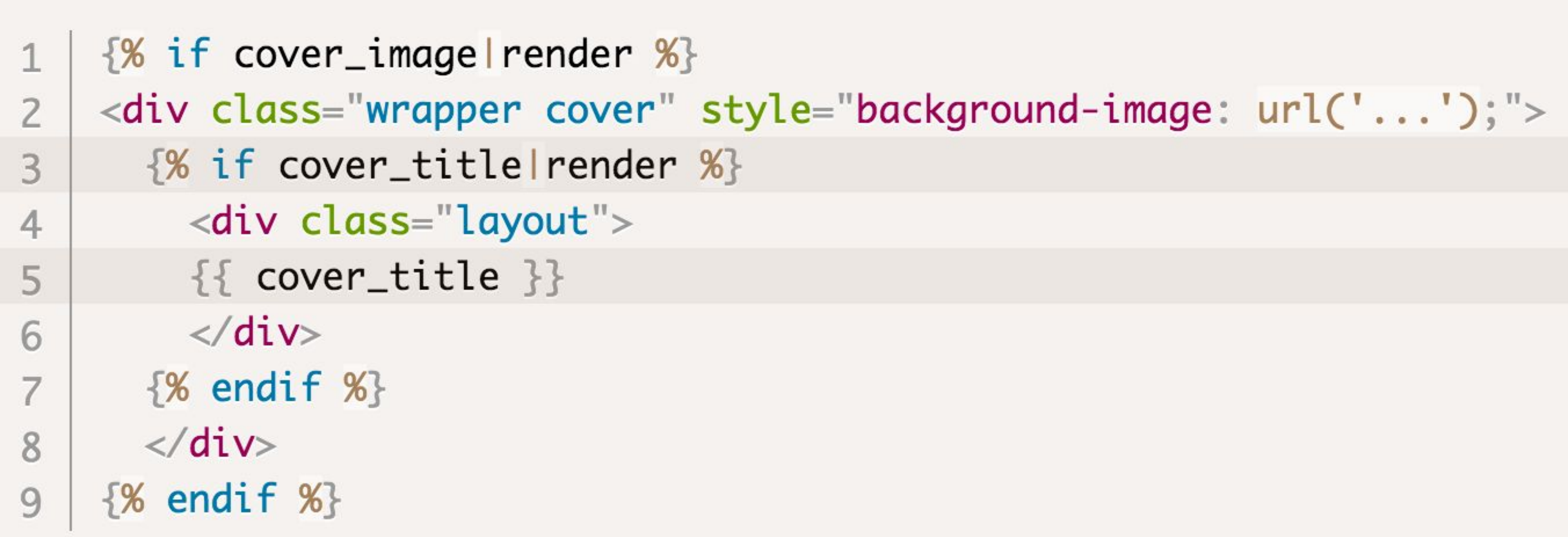
- Modify the *.theme file.
- Implement hook_preprocess_HOOK(&$variables)
- Modify the $variables array. This is a key-value container of information. Keys will become variables in the template.
- Print the variable in the template.

1. How can I pass information from Drupal to Twig?



```php
/**
 * Implements template_preprocess_page().
 */
function nicaragua_preprocess_page(&$variables) {
  if (isset($variables['node']) && $variables['node']->bundle() == 'page') {
    $variables['cover_image'] = $variables['node']->get('field_image')->view('full');
    $variables['cover_title'] = $variables['node']->get('field_teaser')->view('full');
  }
}
```

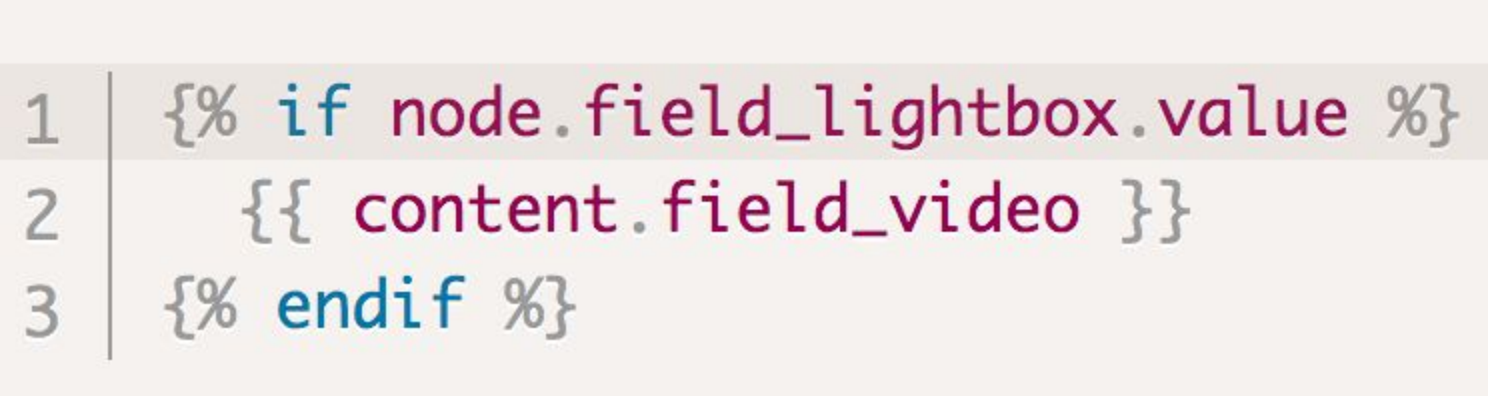
1. How can I pass information from Drupal to Twig?



```
{% if cover_image|render %}
<div class="wrapper cover" style="background-image: url('...');">
  {% if cover_title|render %}
    <div class="layout">
    {{ cover_title }}
    </div>
  {% endif %}
  </div>
{% endif %}
```

2. How can I conditionally render fields based on the content of other fields?

Boolean field

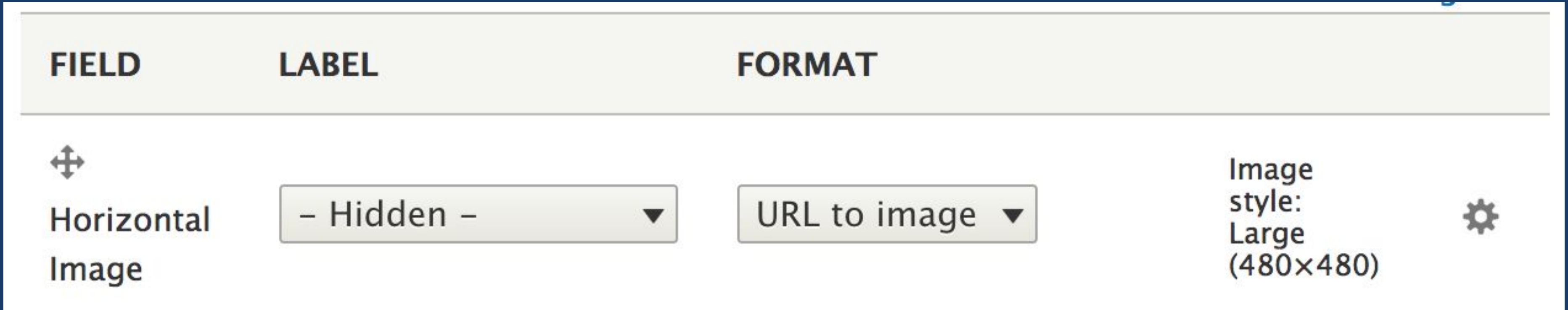


```
{% if node.field_lightbox.value %}
  {{ content.field_video }}
{% endif %}
```

3. How can I render image fields as background images?

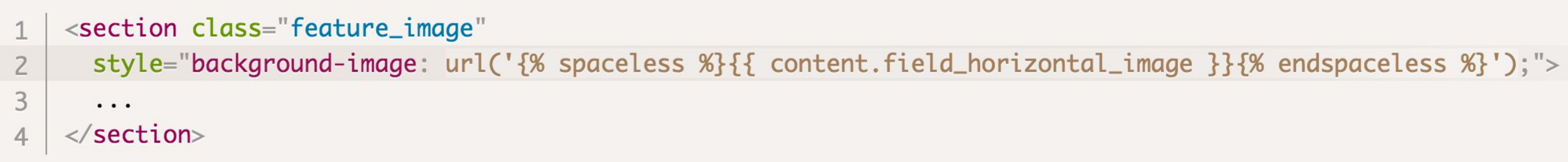
- Configure the image field to link to use the “URL to Image” format.
- Render the variable in the template.

3. How can I render image fields as background images?

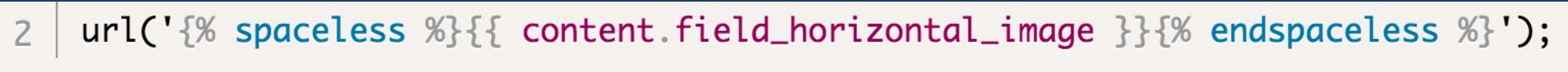


FIELD	LABEL	FORMAT		
✥ Horizontal Image	– Hidden – ▼	URL to image ▼	Image style: Large (480×480)	⚙

3. How can I render image fields as background images?



```
<section class="feature_image"
  style="background-image: url('{% spaceless %}{{ content.field_horizontal_image }}{% endspaceless %}');">
  ...
</section>
```



```
url('{% spaceless %}{{ content.field_horizontal_image }}{% endspaceless %}');
```

3 ½. What about responsive background images?

Article:

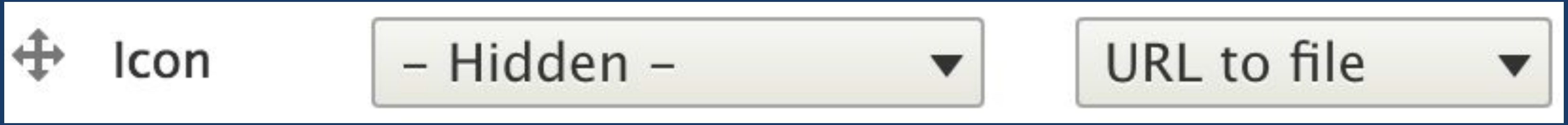
- https://tag1consulting.com/blog/background-image-new-drupal-8-module by Mark Carver

Modules:

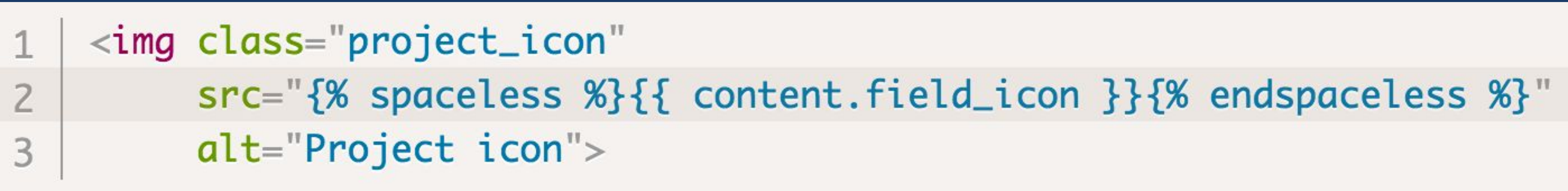
- Background Image
- Background Images Formatter

4. How can I render node content as HTML attributes?

File field (SVG)



Icon | – Hidden – | URL to file



```
<img class="project_icon"
     src="{% spaceless %}{{ content.field_icon }}{% endspaceless %}"
     alt="Project icon">
```

4 ½. Does it have to be so hard to use a SVG image?

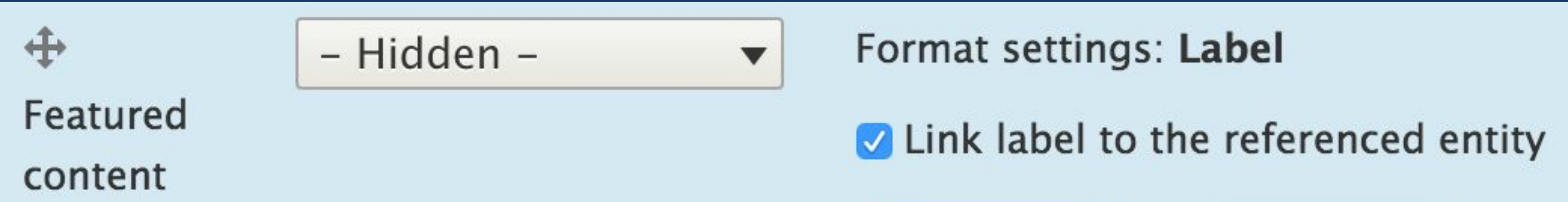
Not any more!

Modules:

- SVG Image
- SVG Formatter
- SVG image field

5. How can I render node content as HTML attributes?

Entity reference field as link



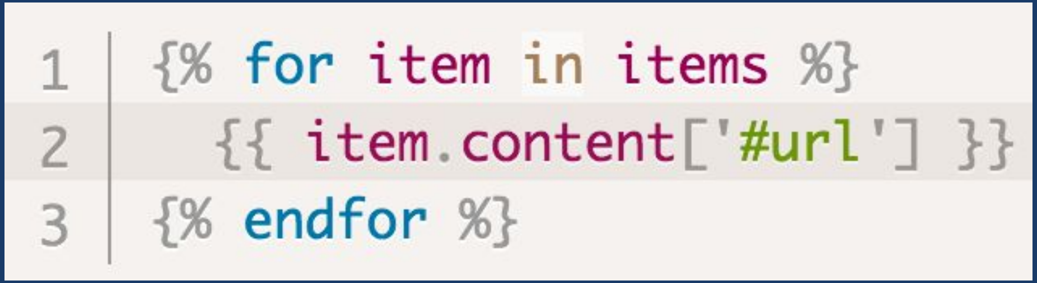
Featured content

– Hidden –

Format settings: **Label**

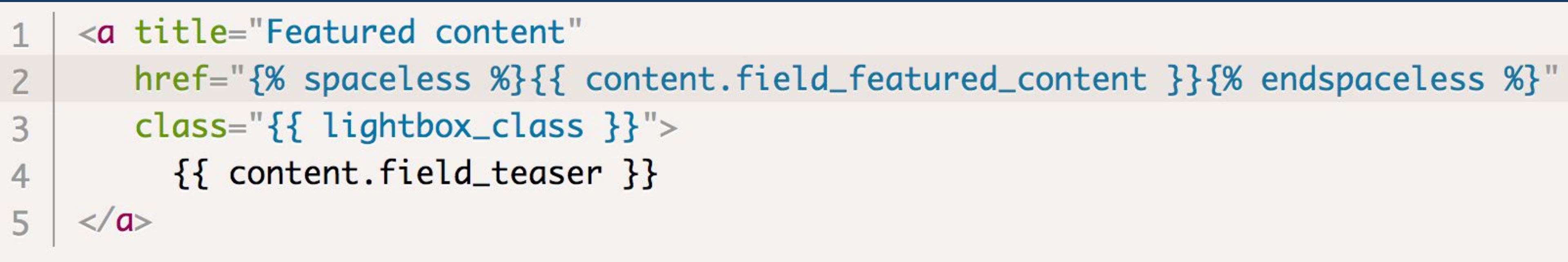
☑ Link label to the referenced entity

field--node--field-featured-content--feature.html.twig



```
{% for item in items %}
  {{ item.content['#url'] }}
{% endfor %}
```

5. How can I render node content as HTML attributes?



```
<a title="Featured content"
   href="{% spaceless %}{{ content.field_featured_content }}{% endspaceless %}"
   class="{{ lightbox_class }}">
     {{ content.field_teaser }}
</a>
```

Contributed modules

- Bamboo Twig
- Component Libraries
- Themers Little Helper
- Twig Clean Debug
- Twig Extender
- Twig Extensions
- Twig Field Value
- Twig Renderable
- Twig Tweak
- Twig Whitespace Collapser
- Twig Xdebug

Want to learn more?

Mastering Twig

by Javier Eguiluz @ DrupalCon Barcelona 2015

https://www.youtube.com/watch?v=4KfkV1T3bhE

Want to learn more?

Drupalnator Themestation:

Understanding Drupal 8's New Theming Layers

by Darryl Norris @ Twin Cities DrupalCamp 2016

https://www.youtube.com/watch?v=Ot--bH9379U

Need support?

#drupal-twig @ freenode.net

https://drupaltwig-slack.herokuapp.com

Thank You!

Let’s continue the conversation on twitter

#DrupalTwigRecipes

Please Provide Feedback

@dinarcon | #DrupalTwigRecipes | mauricio@agaric.com

http://bit.ly/twig-recipes
http://bit.ly/twig-recipes-feedback