

AcQUiA

Drupal Layout and Theming

<https://rodsurl.com/d11lt>

Set up for this morning

1. Download the PDF and zip file from <https://rodsurl.com/d10lt>
2. Go to Content and filter by Hotels.
3. Select all and in bulk actions dropdown (bottom of the page) click Remove content from front page.

The goal here is to have only the Test article appearing on the front/home page.

Action

- ✓ Delete content
- Make content sticky
- Make content unsticky
- Promote content to front page
- Publish content
- Save content
- Remove content from front page**
- Unpublish content
- Update URL alias

AcQUiA

Welcome

Previous Experience

- **Familiar with Drupal site building?**
- **Skilled in HTML and CSS?**
- **Ever used Twig?**

Learning Modules

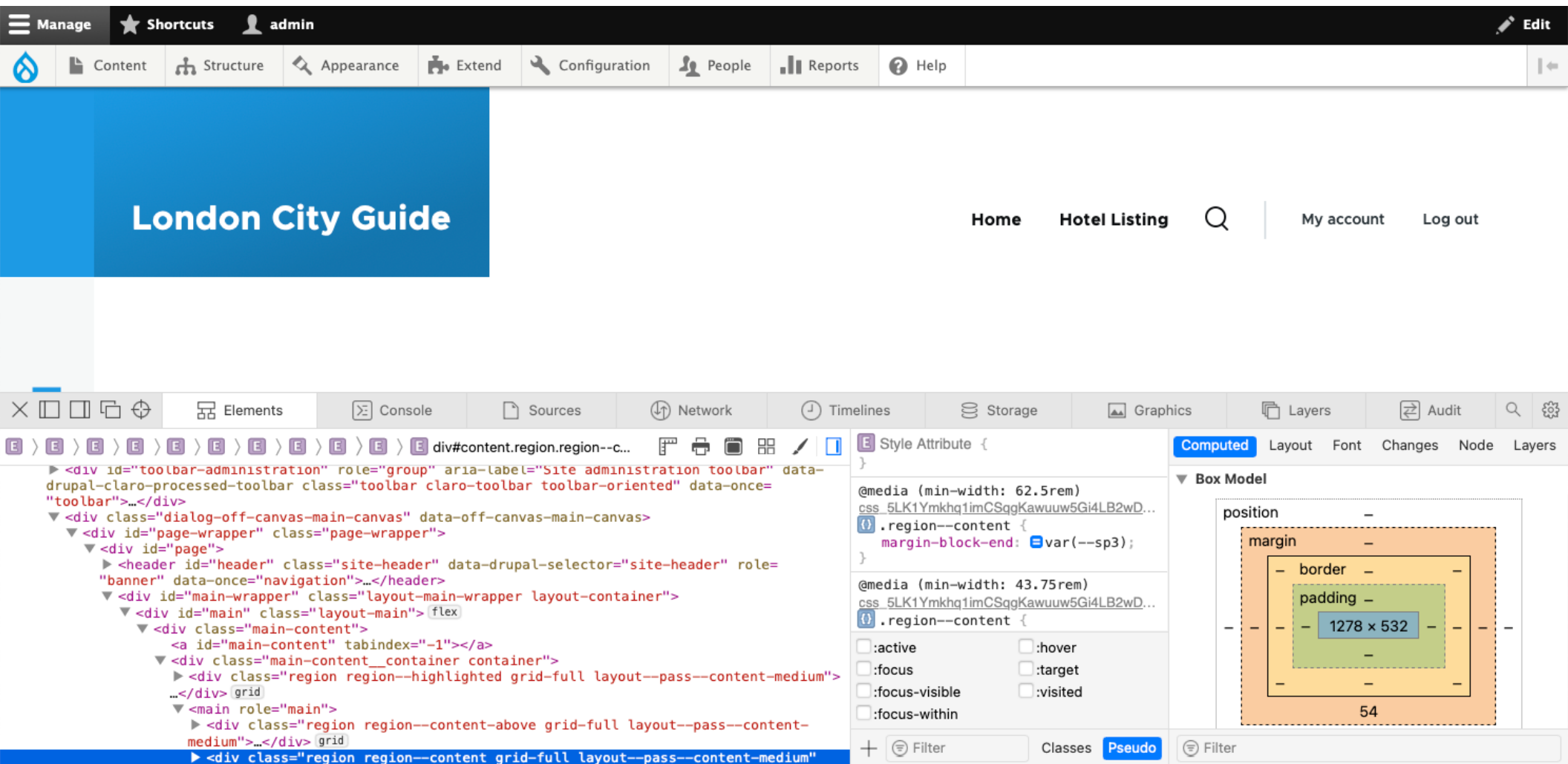
Learning Module 1	Drupal Theming Fundamentals
Learning Module 2	Customizing Your Theme
Learning Module 3	Base Themes and Subtheming
Learning Module 4	Matching a Design Mockup Using Drupal
Appendix	Optional Review: How would you do that in Drupal

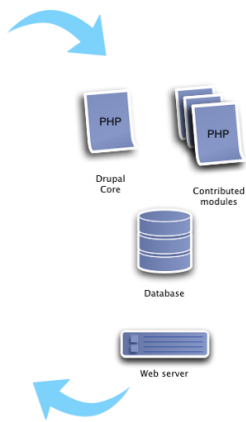
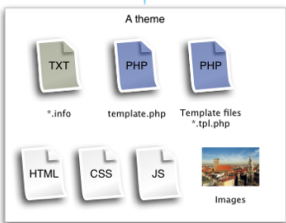
Getting Started

Needed Materials & Setup

- A fresh Drupal instance
- Drupal Layout and Theming Manual / Course files and resources - <https://rodsurl.com/d11lt>
- Lorem Ipsum Generator (<https://lipsum.com>)
- A decent text editor with code highlighting.
 - We'll be using VSCode in the browser today.
 - Windows: Visual Studio Code, Atom, PHP Storm
 - Mac: Visual Studio Code, Coda2, TextWrangler

A Good Browser





B. Discover the elements in the content array

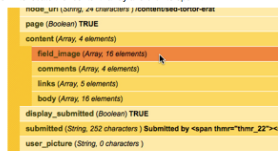
1. Open the template.php file for your theme (created in the previous session). Add the following preprocess function into the template.php file. Make sure the function name "mytheme" matches the name of your theme.

```
function mytheme_preprocess_page(&$variables) {
  kpr($variables);
}
```

You can also kpr() directly to template files with this:

```
<?php kpr(get_defined_vars()); ?>
```

2. Save the file and clear your cache. Like the .info file, the template.php file is cached. When you add new functions you need to clear your cache.
3. Visit an article to which you have uploaded an image. This will generate a click-able display of your node content.
4. Scroll down to locate the "content" array and click to open it.



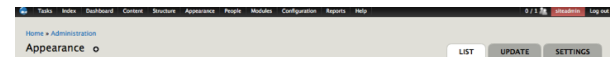
Now you can see what is being output.

C. Control the display with hide() and render()

Comments, links and images can be excluded from the output initially, then rendered afterwards. There you can control how they are displayed.

1. Open the default node.tpl.php. Locate the start of div class="content" and replace the remainder of the node template with this code.

```
<div class="content"><?php print $content_attributes; ?>
<?php
// We hide the comments and links now so that we can
render them later.
hide($content['comments']);
hide($content['links']);
hide($content['field_image']);
print render($content);
?>
```



Set and configure the default theme for your website. Alternative themes are available.

[Install new theme](#)

ENABLED THEMES



My theme
A great theme
[Settings](#) | [Disable](#) | [Set default](#) | [Style guide](#)

DISABLED THEMES



Blue Masters 7.x-1.2
Theme based on Blue Masters PSD template
[Enable](#) | [Enable and set default](#)



Busy 7.x-1.4
The ultimate Drupal business theme.
[Enable](#) | [Enable and set default](#)



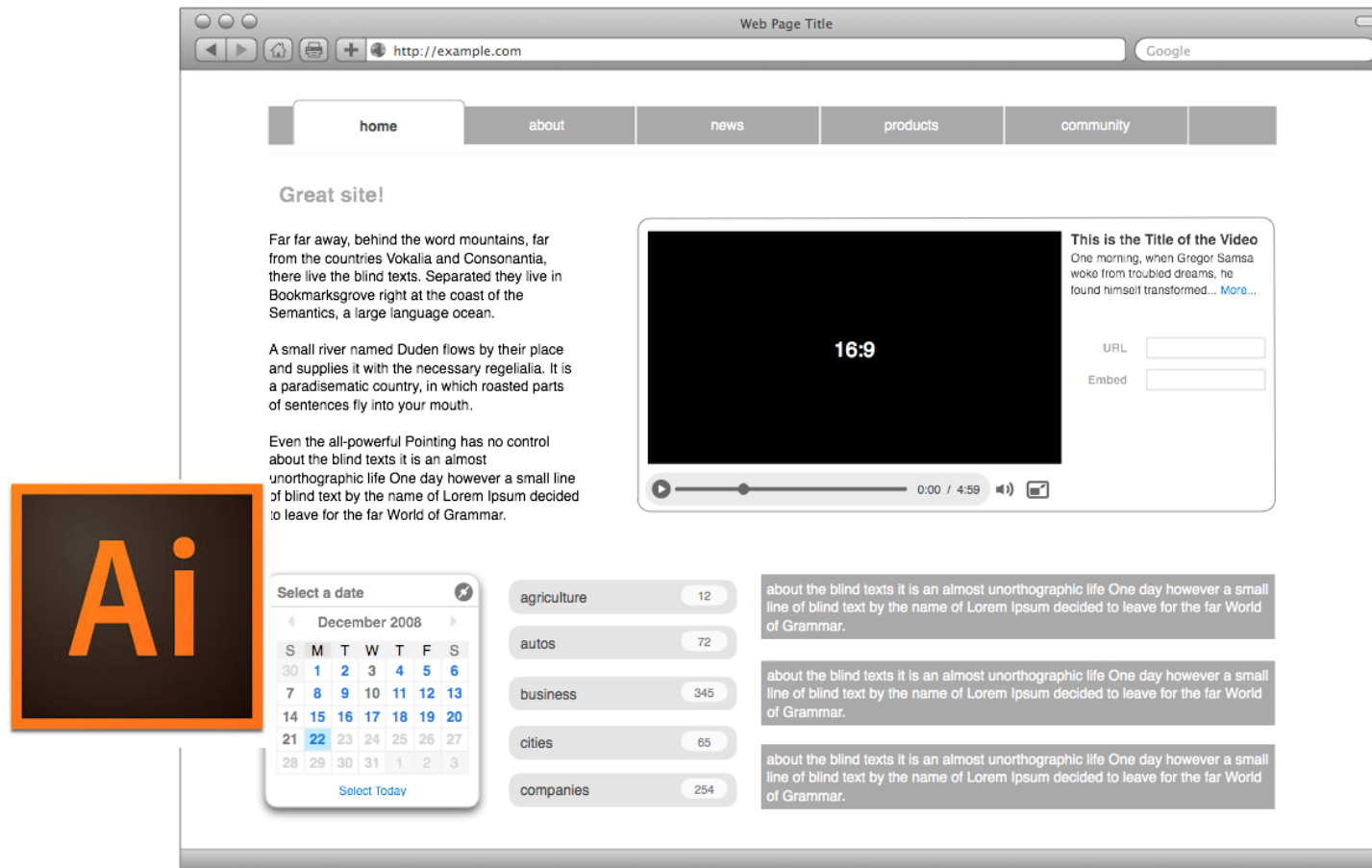
Framework 7.x-3.6
Framework is a blank canvas for theme developers. Use Framework as a starting point to facilitate your theme development.

Drupal Theming Fundamentals

- **Learning
Module 1**



Start with a Great Design



Develop Perfect Markup

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>My Perfect Markup!</title>
    <meta charset="utf-8">
    <meta name="description" content="This is the best HTML ever.">
    <link rel="Shortcut Icon" type="image/ico" href="img/favicon.png">
    <link rel="stylesheet" href="css/reset.css" type="text/css"
      media="screen">
    <link rel="stylesheet" href="css/style.css" type="text/css"
      media="screen">
    <body>
      <div id="wrap">
        <header>
          <div class="inside">
            <script type="text/javascript"><h1>Best <span>Ever</span></h1></script>
            <script type="text/javascript" src="http://example.com/js/main.js"></script>
            <nav>
              <ul>
                <li><a href="#">Home</a></li>
                <li><a href="#">About</a></li>
                <li><a href="#">Contact</a></li>
              </ul>
            </nav>
          </div>
        </header>
        <article>
          <section>
```

Integrate with “Back End”

```
<div class="userprofile">
{exp:query sql="SELECT username, screen_name, email, location FROM exp_members WHERE group_id = '1'"}
    {username}
    {screen_name}
    {email}
    {location}
{/exp:query}
</div>

<div class="categories">
{exp:channel:categories channel="mychannel"}
    <a href="{path=mychannel}">{category_name}</a>
{/exp:channel:categories}
</div>
```

Uh oh...

```
<div class="userprofile">
{exp:query sql="SELECT screen_name, email, location FROM exp_members WHERE group_id = '1'"}
    {screen_name}
    {email}
    {location}
{/exp:query}
</div>

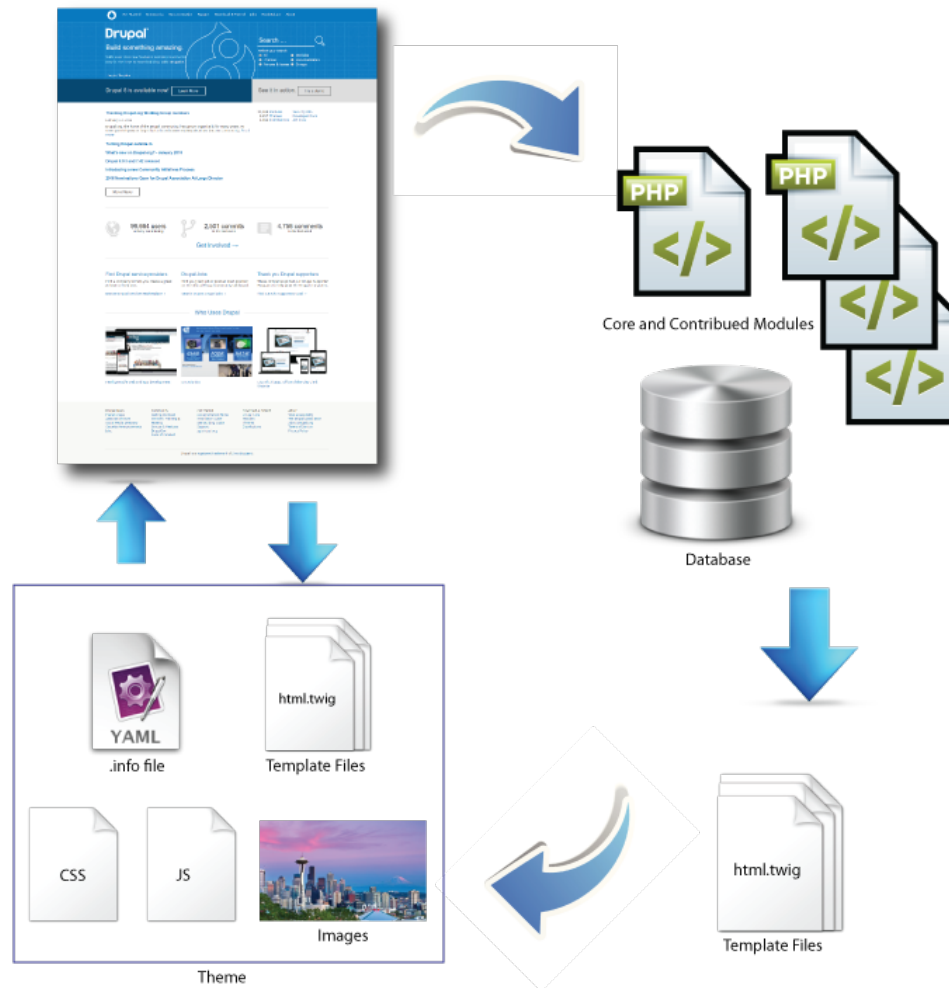
<div class="categories">
{exp:channel:categories channel="mychannel"}
    <h2>{category_name}</h2>
    {if category_description}
        <p>{category_description}</p>
    {/if}
{/exp:channel:categories}
</div>
```

Alter query

Alter markup

Add logic

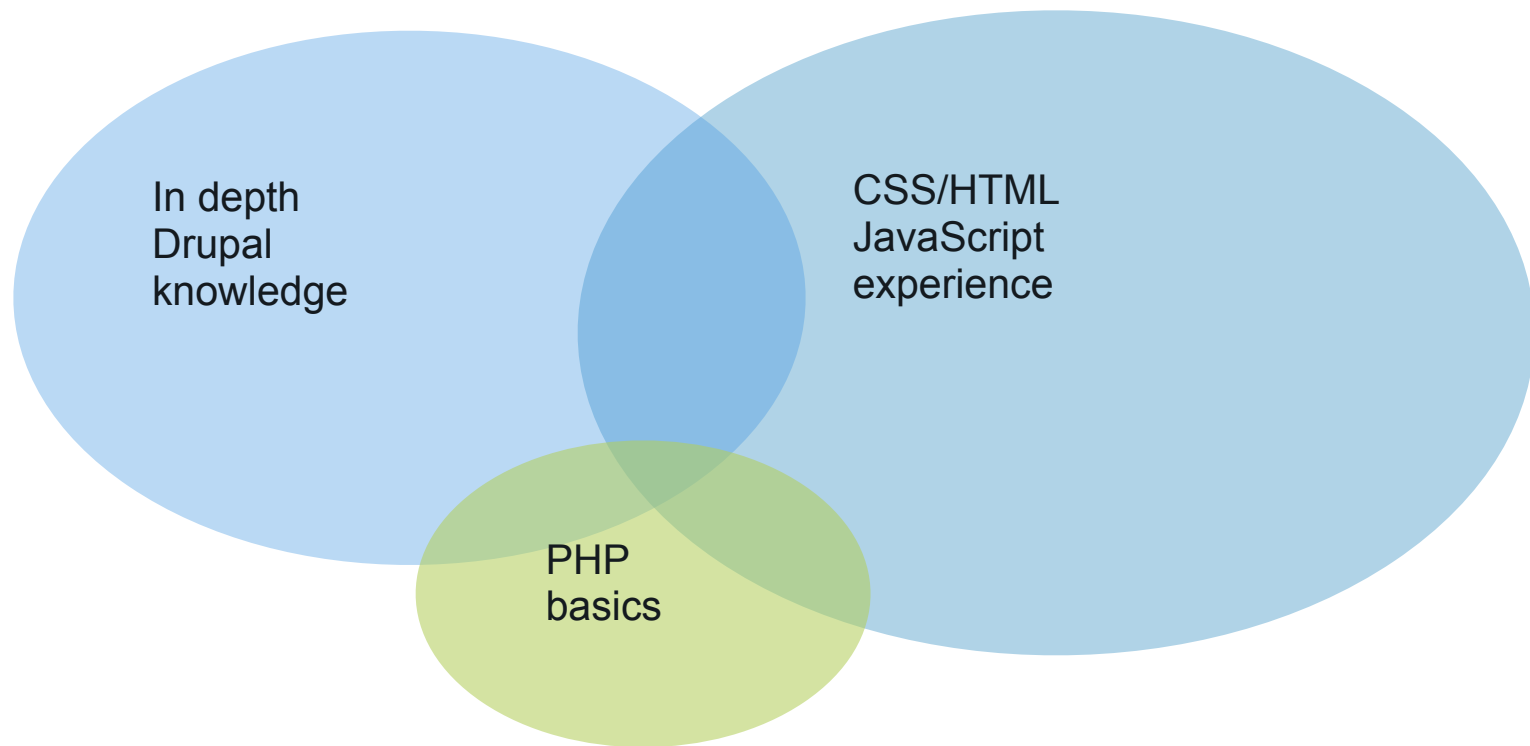
How Does Drupal Theming Work?



Chain Of Command



What are Themers?



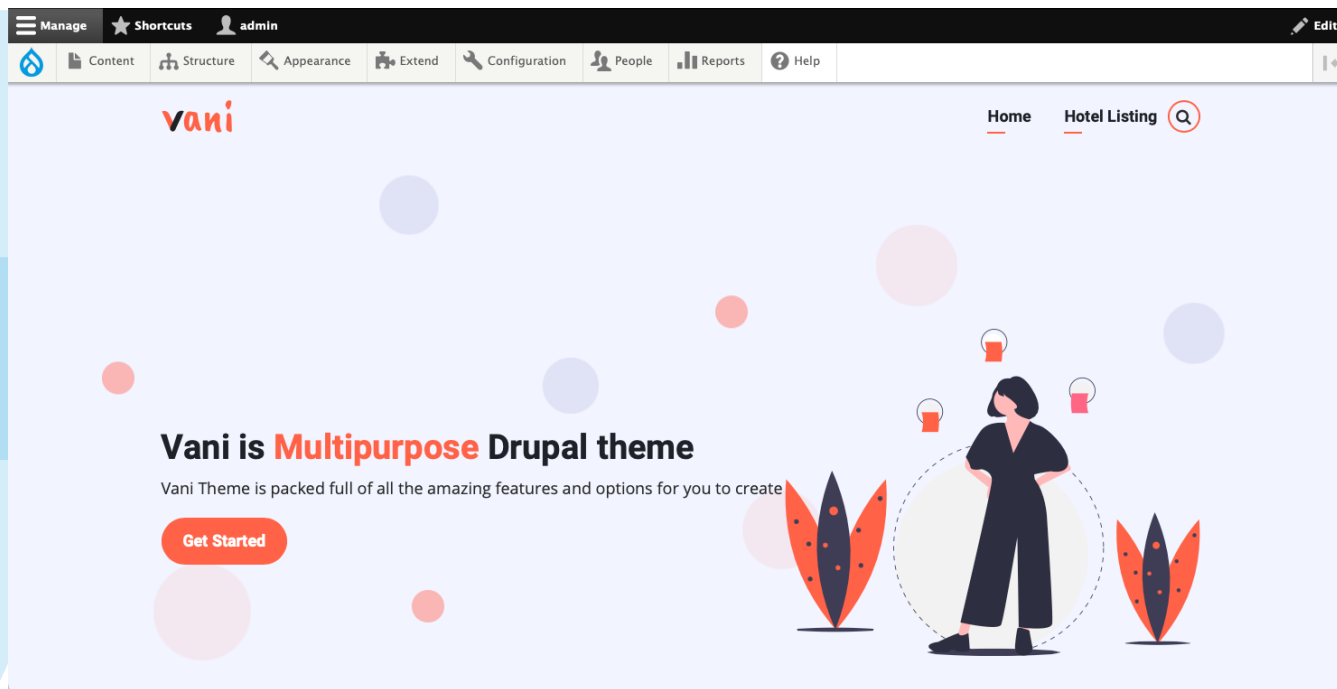
What About You?

- **What kinds of roles do you typically take?**
- **What is your team like?**
- **Think about skills you want to develop.**
- **What previous systems did you use?**

Theming Comparison

- In Python it's called “templating”
- In Joomla it's “template development”.
- In WordPress and Drupal it's “theming”.

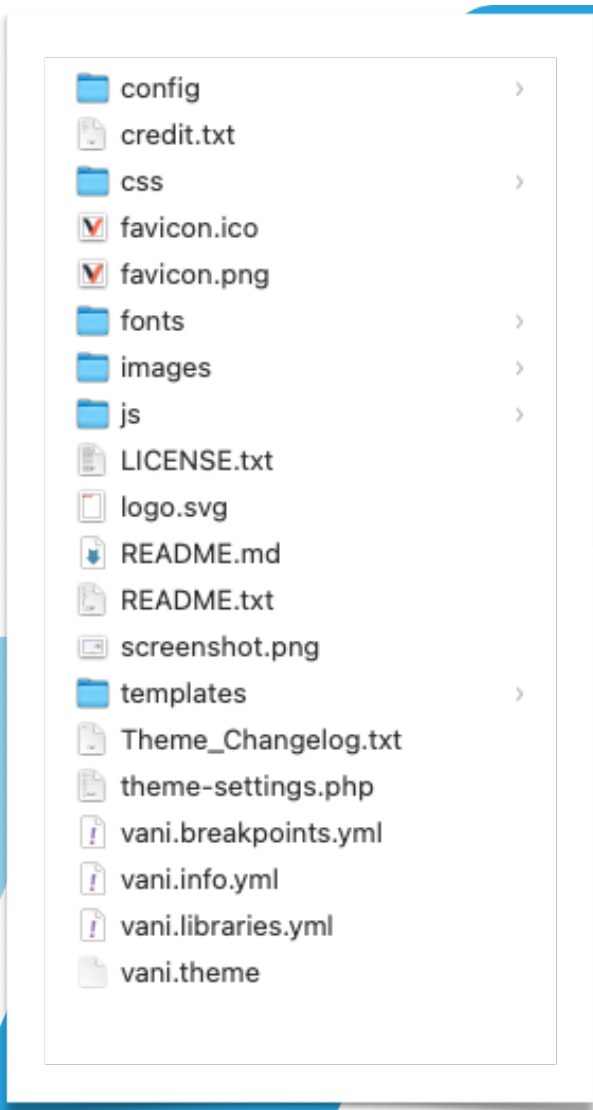
Example Theme: Vani



<https://drupal.org/project/vani>

Vani Theme Folder

- **Contains**
 - **.info.yml config file**
 - **Screenshot, logo**
 - **CSS files**
 - **Images**
 - **Twig templates**

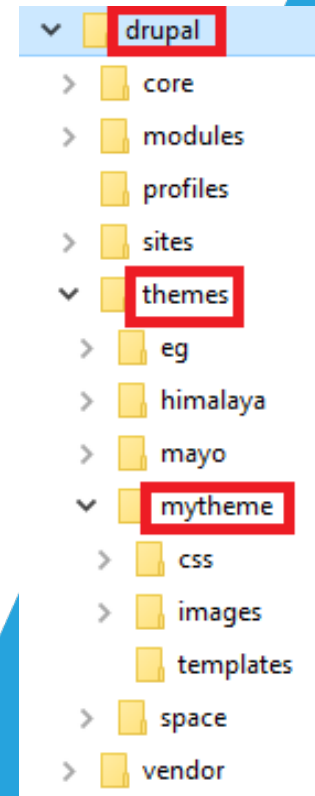


A screenshot of a file explorer window displaying the contents of the Vani Theme folder. The list includes folders for 'config', 'css', 'fonts', 'images', and 'js', each followed by a right-pointing arrow. It also lists several files: 'credit.txt', 'favicon.ico', 'favicon.png', 'LICENSE.txt', 'logo.svg', 'README.md', 'README.txt', 'screenshot.png', 'Theme_Changelog.txt', 'theme-settings.php', 'vani.breakpoints.yml', 'vani.info.yml', 'vani.libraries.yml', and 'vani.theme'. Each file is preceded by its respective icon (e.g., folder icon for directories, document icon for text files, and image icon for images).

- config
- credit.txt
- css
- favicon.ico
- favicon.png
- fonts
- images
- js
- LICENSE.txt
- logo.svg
- README.md
- README.txt
- screenshot.png
- templates
- Theme_Changelog.txt
- theme-settings.php
- vani.breakpoints.yml
- vani.info.yml
- vani.libraries.yml
- vani.theme

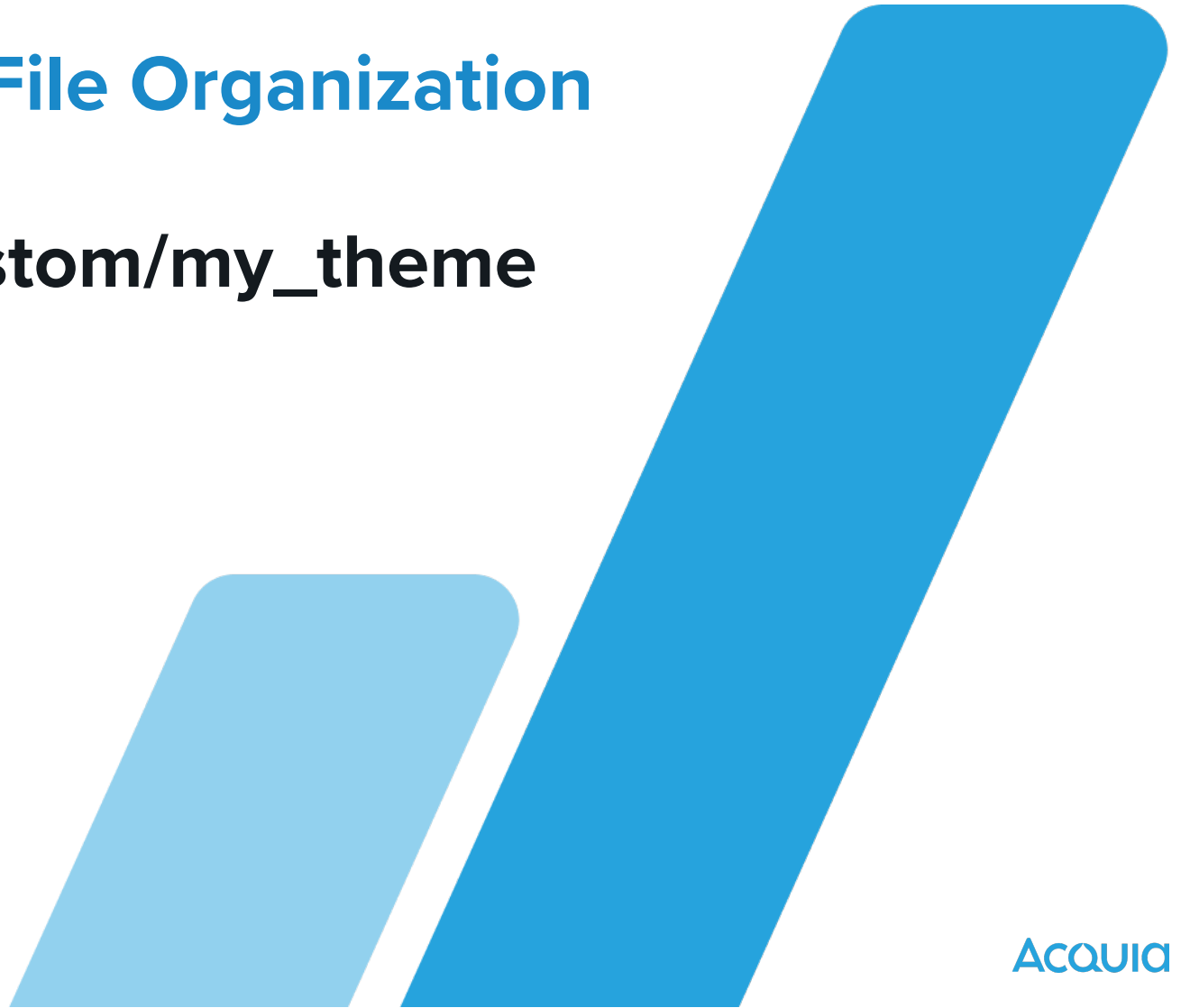
Theme Folder in Drupal Hierarchy

- **Location in file hierarchy:**
 - **Drupal core folder/themes/**



Custom Theme File Organization

- **`/themes/custom/my_theme`**



Core Theme File Organization

- **`/core/themes/stable9`**



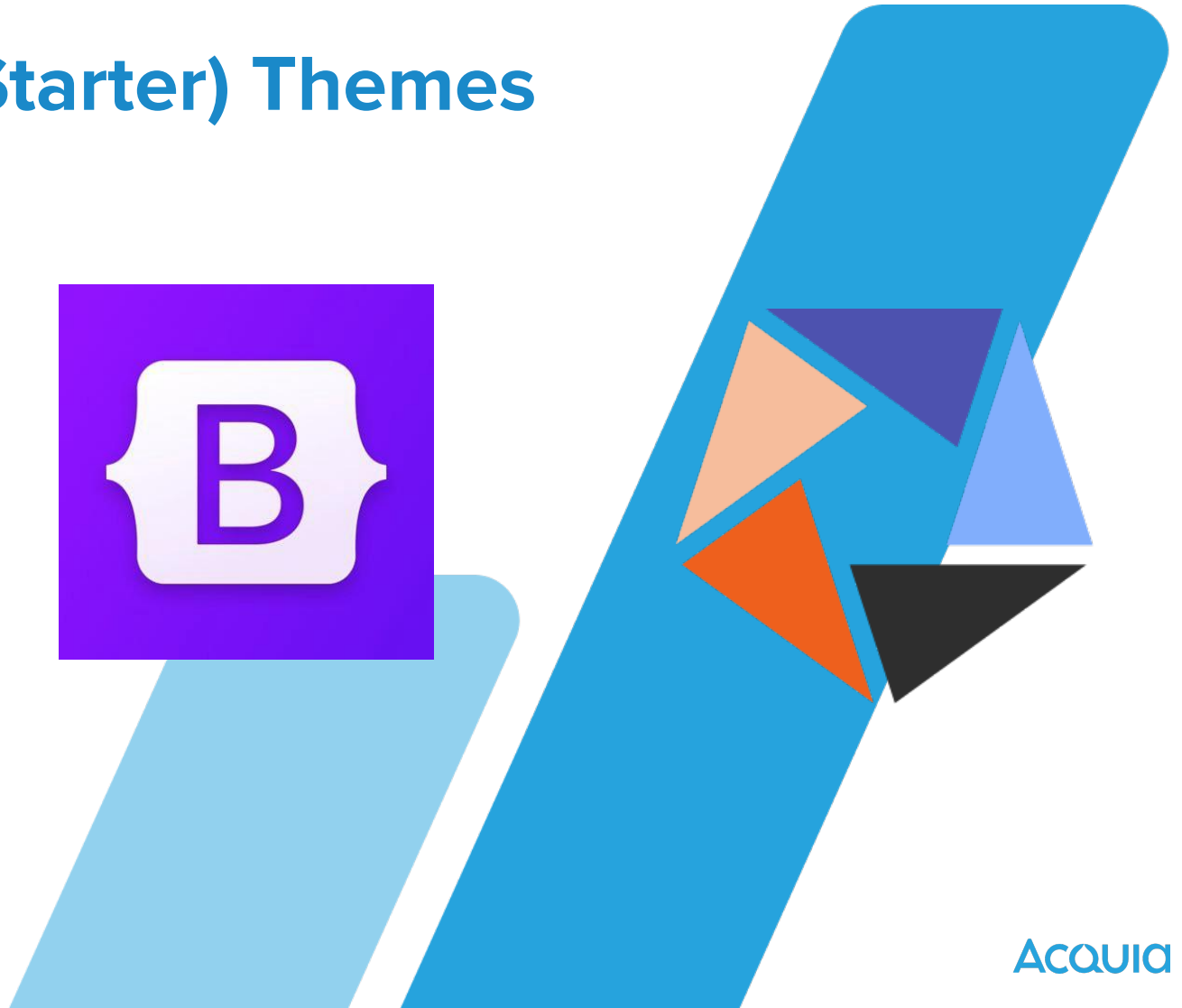
Types of Themes

Core themes	Drupal core comes with a few themes. These are suitable for very basic sites.
Contributed Standard themes	Free themes that have been contributed back to the Drupal Community. http://drupal.org/project/themes
Contributed Starter/Base themes	Base or Starter themes are contributed themes designed to be used as a starting point for a custom subtheme.
Custom themes	Most sites require a custom look and feel. These are often created as subthemes of a <i>starter</i> or <i>base theme</i> .
Contributed Administration themes	Themes that are displayed only in administration sections of a site
Premium themes	Premade themes available for purchase.

Finding & Evaluating Contributed Themes

- **Contributed Theme Considerations**
 - Two types of themes, visitor-facing, and admin-facing
 - Base themes may not be complete enough for site builders new to Drupal
- https://www.drupal.org/project/project_theme

Example Base (Starter) Themes



ACQUIA

Project page

Download & Extend

[Drupal Core](#) [Distributions](#) [Modules](#) [Themes](#) [General projects](#)

Bootstrap

[View](#) [Version control](#) [Automated testing](#)

“Sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. Bootstrap has become one of the most popular front-end frameworks and open source projects in the world.”

This base theme bridges the gap between Drupal and the [Bootstrap Framework](#).

Bootstrap 3 for Drupal

Features

- [jQuery CDN](#) for “out-of-the-box” styling and faster page load times.
- [Bootswatch](#) theme support, if using the CDN.
- Glyphicons support via [Icon API](#).
- Extensive integration and template/preprocessor overrides for most of the [Bootstrap Framework](#) CSS, Components and JavaScript
- Theme settings to further enhance the Drupal Bootstrap integration:
 - [Breadcrumbs](#)
 - [Navbar](#)
 - [Popovers](#)
 - [Tooltips](#)
 - [Wells](#) (per region)

Documentation

Visit the project's [official documentation site](#) or the markdown files inside the `./docs` directory.

Supported modules

Drupal 8

- [Bootstrap Layouts](#)
- [Views Bootstrap](#)

Drupal 7

- [Bootstrap Core](#)
- [jQuery Update](#)
- [Icon API](#)
- [Picture](#)
- [Views](#) (partial support)
- [Webform](#) (partial support)

Help support Drupal Bootstrap and the entire Drupal community

This project is part of Drupal's contributed ecosystem, both of

★ 174

Maintainers

[wundo](#) [shelane](#)

Issues for Bootstrap

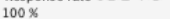
To avoid duplicates, please search before submitting a new issue.

[Search](#)

Advanced search

All issues
99 open, 3 RTBC, 2,601 total
Bug report
56 open, 2 RTBC, 1,168 total

Statistics

New issues  0
Response rate  100 %
1st response  151 hours
Open bugs  52
Participants  7
2 year graph, updates weekly

Resources

[Home page](#)
[Read license](#)
[Read complete log of changes](#)
[Try out a demonstration](#)
<http://drupal-bootstrap.org>
[View project translations](#)
[Projects that extend this](#)

Development

[Source code](#)
[Commit log](#)

Who maintains this?

What are current issues?

Download links

Similar projects?

Documentation?

Project information

Project information

Ecosystem: Barrio Bootstrap 5 Theme



40,084 sites report using this theme

Created by [hatuhay](#) on 17 May 2013, updated 9 September 2024



Stable releases for this project are covered by the [security advisory policy](#).
Look for the shield icon below.

Issues for Barrio Bootstrap 5 Theme

To avoid duplicates, please search before submitting a new issue.

Search

[Advanced search](#)

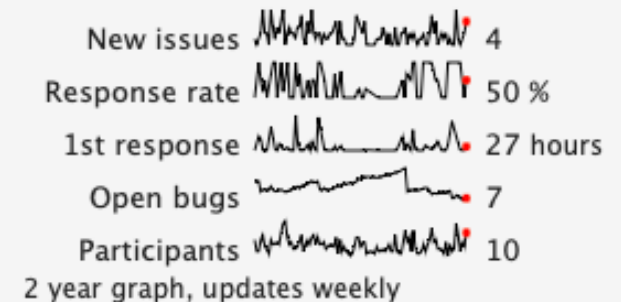
All issues

59 open, 534 total

Bug report

12 open, 283 total

Statistics



Versions

Releases

5.5.19  released 6 December 2024

Works with Drupal: ^10.3 || ^11.0

✓ Recommended by the project's maintainer.

Component based, Drupal 10.3+, 11, Bootstrap 5

Install: `composer require 'drupal/bootstrap_barrio:^5.5'` 

Development version: [5.5.x-dev](#) updated 6 Dec 2024 at 15:28 UTC

5.1.12  released 4 September 2024

Works with Drupal: ^9.4 || ^10 || ^11

Install: `composer require 'drupal/bootstrap_barrio:^5.1'` 

Development version: [5.x-dev](#) updated 4 Sep 2024 at 02:10 UTC

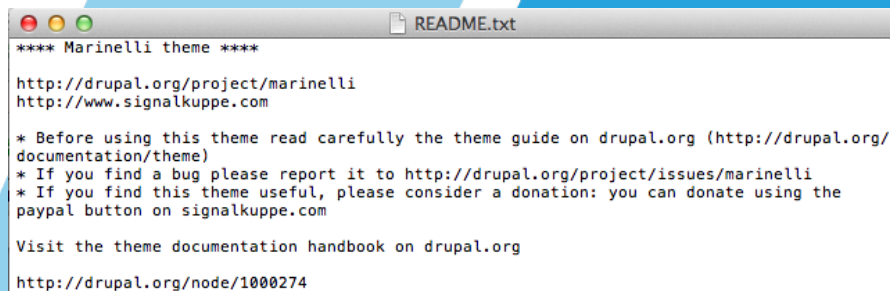
Installing Themes

— Using Composer

- Run `composer require drupal/{theme_name}`
- ie: `composer require 'drupal/bootstrap_barrio:^5.5'`

Tip: README.txt

- **Key information in README.txt**
 - When working with a new theme, always review for documentation, notes about requirements, how to install, how to report bugs, etc.

A screenshot of a text editor window titled 'README.txt'. The window contains the following text:

```
**** Marinelli theme ****  
  
http://drupal.org/project/marinelli  
http://www.signalkuppe.com  
  
* Before using this theme read carefully the theme guide on drupal.org (http://drupal.org/documentation/theme)  
* If you find a bug please report it to http://drupal.org/project/issues/marinelli  
* If you find this theme useful, please consider a donation: you can donate using the paypal button on signalkuppe.com  
  
Visit the theme documentation handbook on drupal.org  
http://drupal.org/node/1000274
```

Exercise:

Install a Theme via Composer

Go to <https://www.drupal.org/project/vani> to see the page for the Vani theme.

Copy the composer command.

Return to your Terminal and paste the command into the terminal.

Go to your Drupal site. Click on Appearance > Install and Set as Default for Vani.



Static and Configurable Themes

- **Change through configuration:**

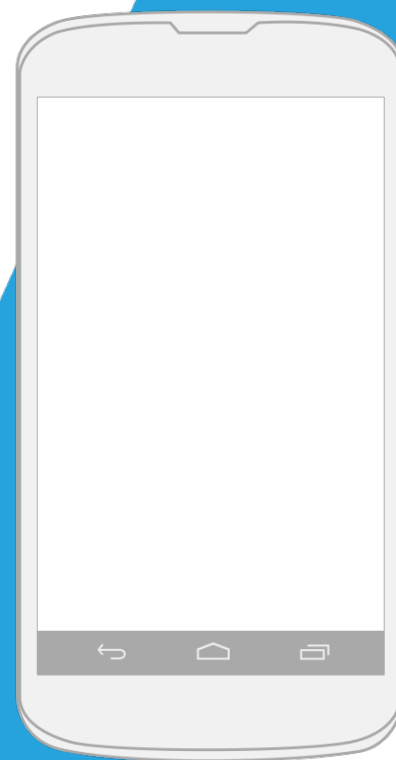
- Default settings only

- **Change through configuration:**

- theme images
 - layout
 - media queries
 - external font libraries
 - etc

Responsive Theming Considerations

- The resizability of your layout and grid system
- Navigation usability on devices with large or small screens
- Slideshows and scripts that hide or resize when viewed on small screens
- Tables that also resize or collapse when viewed from small devices
- Learn more about Drupal's Mobile-First Design Initiative - <https://www.drupal.org/docs/8/mobile/responsive-web-design>



The Contents of a Simple Theme



css



images



templates



mytheme.info.yml



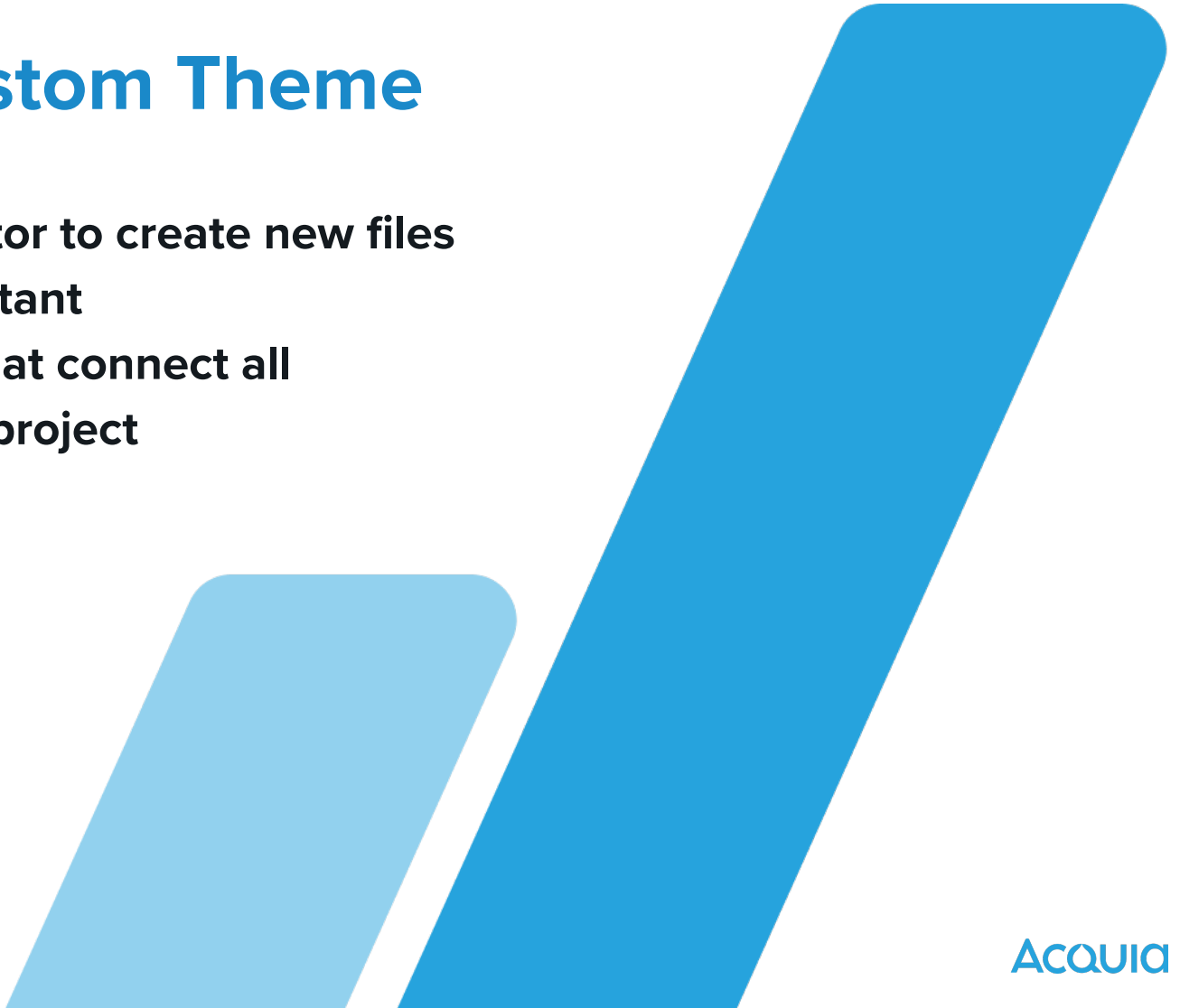
mytheme.libraries.yml



screenshot.png

Let's Make a Custom Theme

- You'll need your text editor to create new files
- Project names are important
 - They're the “key” that connect all components of the project
 - Must be unique



.info.yml File

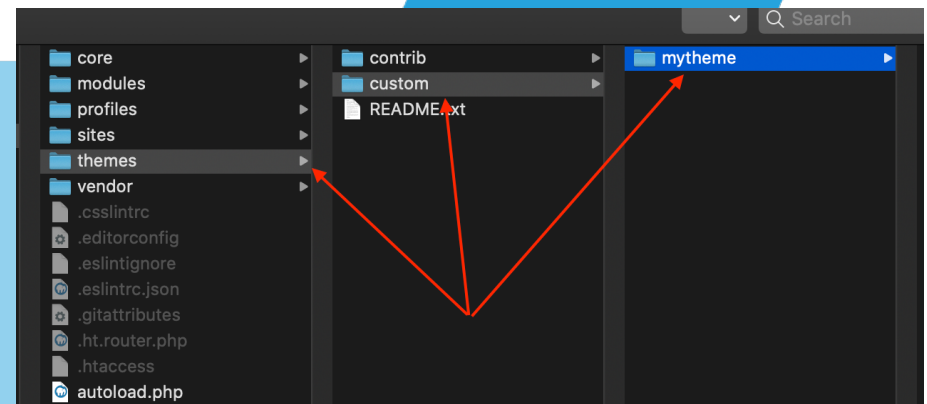
- **A .info.yml file is the only required file of a theme**
- **It makes declarations to Drupal, specifying:**
 - the theme name label
 - type of project (module or theme)
 - core compatibility
 - regions and feature overrides
 - location of CSS and other needed files
- **Optional values not specified will use Drupal's default**

You'll Start by Creating your Folder



Exercise: Make the .info.yml File for Your First Theme

**Create a mytheme folder
in the /themes/ directory
Start a mytheme.info.yml
file in your text editor,
save to your project
folder**



Contents of mytheme.info.yml

```
name: My Great Theme
type: theme
description: This is my first theme project
core_version_requirement: ^10 || ^11
base theme: stable9
```

Exercise:

Add a Logo and Screenshot

**Copy screenshot and logo
from course asset folder**

Declare in mytheme.info.yml

**Visit Manage > Structure > Block layout
and ensure the Site Branding block is in
the Header region**

**Configure to turn off Site name and Site
slogan**

THIS IS MY LOGO

ACQUIDTM

ACQUID

Updated Contents of mytheme.info

```
name: My Great Theme
type: theme
description: This is my first theme
project
core_version_requirement: ^10 || ^11
base theme: stable9
screenshot: mytheme.png
```

Regions

- The areas you assign blocks to
- Drupal has default regions that it uses if you do not declare **ANY** custom regions in `.info.yml`
- Declare **ONE** custom region, you need to declare all
- **Three files help you customize regions:**
 - Declare in the `.info.yml`
 - Render `page.html.twig` template
 - Style with `css`

Default Regions

- **sidebar_first**: Items for the first sidebar.
- **sidebar_second**: Items for the second sidebar.
- **content**: The main content of the current page.
- **header**: Items for the header region.
- **primary_menu**: Items for the primary menu region.
- **secondary_menu**: Items for the secondary menu region.
- **footer**: Items for the footer region.
- **highlighted**: Items for the highlighted content region.
- **help**: Dynamic help text, mostly for admin pages.
- **breadcrumb**: Items for the breadcrumb region.

Hidden Regions

- **Two hidden regions**
 - page_top
 - page_bottom
- **Not available to for blocks, but modules and Drupal system messages might utilize them**

Note About Region Order

- The regions you define are displayed in **Manage > Structure > Block layout**
- This **Block layout** list will match the labels you assign and the order you used in your **.info.yml**

Exercise: Adding Regions

Open mytheme.info.yml.
Add the code to the right.
Save and clear the cache.
Go to the Block Layout page.
You'll only see the three visible
regions you declared.
Click Demonstrate block regions.
Notice it reflects your regions!

```
regions:  
  header: Header  
  content: Content  
  footer: Footer  
  page_top: Page top  
  page_bottom: Page bottom
```

Exercise: Add the .libraries.yml file to your theme

Create sub-folder in your theme called css

Add relevant css stylesheet files provided in the css folder

Create a mytheme.libraries.yml text file

Add the libraries declarations to the file

Do not use tabs, and watch spaces

```
1 global-css:  
2   version: VERSION  
3   css:  
4     theme:  
5       css/elements.css: {}  
6       css/layout.css: {}  
7       css/navigation.css: {}  
8       css/print.css: { media: print }  
9
```

Exercise: ...then edit .info.yml

**Add this to
mytheme.info.yml
(spacing is important!):**

```
6 libraries:  
7   - mytheme/global-css
```

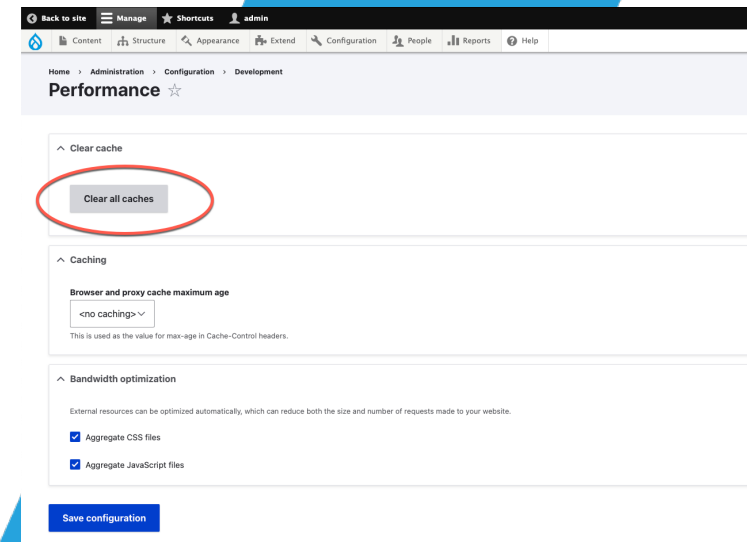
Exercise: Finally, clear the cache!

After updating `.info.yml` files
and `.libraries.yml` files, always clear your
cache

**Manage > Configuration > Development
> Performance > Clear all caches**

Check out the effects added by your new
CSS

Another way to do this is in the terminal.
Type `drush cr` and press enter.



Using Remote Assets

- **Content Delivery Networks**
- **Remote CSS**
- **Webfonts**



Using JavaScript Assets

- You can also define JavaScript assets for your theme
- Make a container for them in `.libraries.yml`
- Syntax is similar to CSS. Example:

```
picturefill:  
  remote: https://github.com/scottjehl/picturefill  
  version: "3.0.1"  
  license:  
    name: MIT  
    url: https://github.com/scottjehl/picturefill/blob/3.0  
    gpl-compatible: true  
  js:  
    assets/vendor/picturefill/picturefill.min.js: { weight:
```

Declaring Dependencies

- **Drupal provides assets like jQuery and normalize.css**
 - asset libraries available for your theme
 - does not load scripts by default
 - you selectively loads libraries
- **Core asset location: /core/assets/vendor**
- **Syntax for loading core assets: - core/assetname**
- **Syntax for other installed assets: - project/asset**

```
global-js:  
dependencies:  
  - core/jquery
```

Exercise: Adding a Webfont

Open mytheme.libraries.yml

Add the following after the end of global-css:

```
10 pacifico:  
11   license:  
12     name: SIL Open Font License 1.1  
13     url: http://scripts.sil.org/cms/scripts/page.php?site\_id=nrsi&id=OFL  
14   css:  
15     theme:  
16       //fonts.googleapis.com/css?family=Pacifico: { type: external }  
17     css/styles.css: {}
```

Exercise: Updating .info.yml

Open
mytheme.info.yml
Update the list of
libraries to look
like this:

```
name: My Great Theme
type: theme
description: This is my first theme project
core_version_requirement: ^9.5 || ^10
base theme: stable9

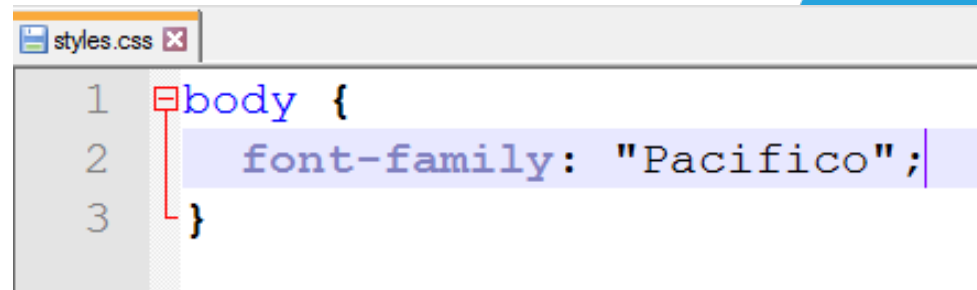
screenshot: mytheme.png

regions:
  header: Header
  content: Content
  footer: Footer
  page_top: Page top
  page_bottom: Page bottom

libraries:
  - mytheme/global-css
  - mytheme/pacifico
```

Exercise: Adding styles.css

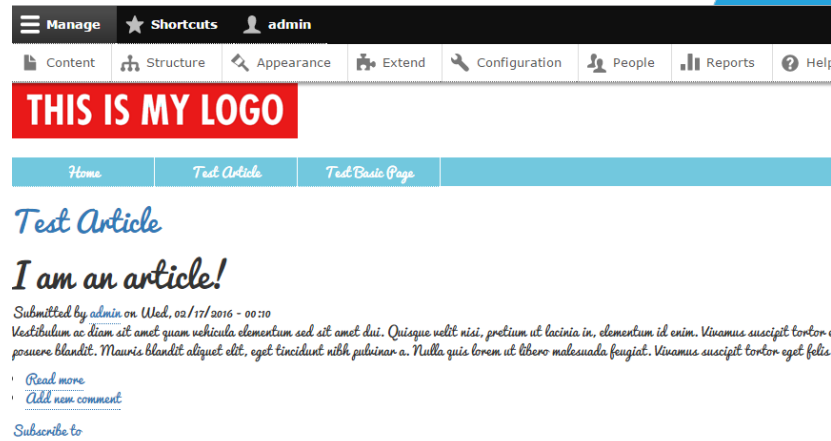
Create styles.css in the /css/ folder
Add the following CSS:

A screenshot of a code editor window titled 'styles.css'. The editor shows three lines of CSS code. Line 1: 'body {' with a red squiggly line under 'body'. Line 2: 'font-family: "Pacifico";' with a blue highlight. Line 3: '}' with a red squiggly line under '}'.

```
1 body {  
2   font-family: "Pacifico";  
3 }
```

Exercise: Result

Clear the cache
Visit your page!



Advanced .info.yml Options

- **libraries-extend**

```
libraries-extend:  
  core/drupal.user:  
    - mytheme/user1  
    - mytheme/user2
```

- **libraries-override**

```
libraries-override:  
  # Replace an entire library.  
  core/drupal.collapse: mytheme/collapse
```

- **stylesheet-
remove**

```
stylesheets-remove:  
  - core/assets/vendor/normalize-css/normalize.css
```

Adding Breakpoints

- Are used in responsive design
- Consist of a label and a media query
- Media queries encode the breakpoints, allow themers to implement different ways of displaying content
- Breakpoints are defined in a `.breakpoints.yml` file
- Breakpoint name has syntax of `projectname.descriptor`

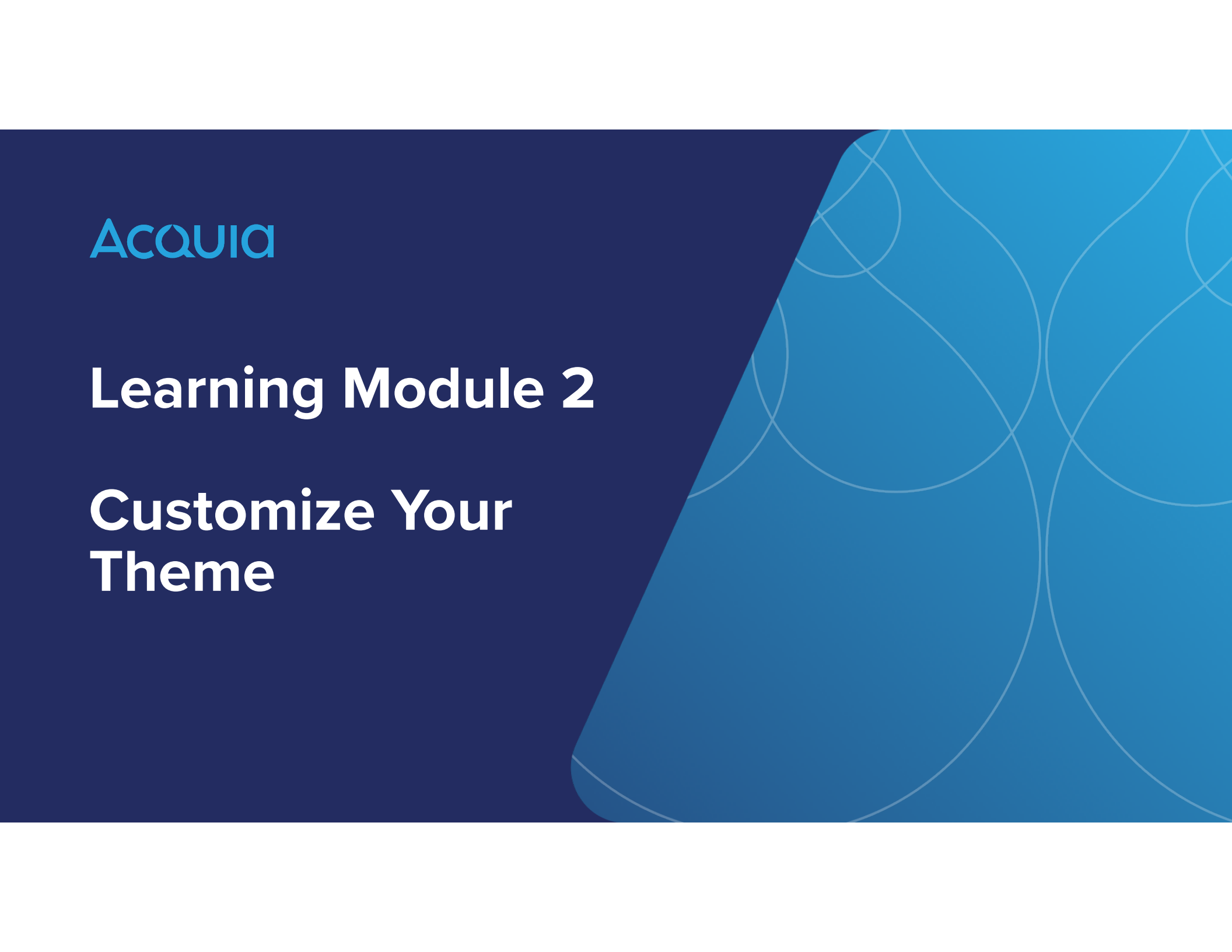
Exercise: Discover Breakpoints from Installed Themes

Navigate to Manage > Extend and ensure the core Responsive Image module is enabled.

Navigate to Manage > Configuration > Media > Responsive image styles > +Add responsive image style, and fill in the form:

Label: My Theme Images
Breakpoint group: Bartik

```
mytheme.mobile:  
  label: mobile  
  mediaQuery: ''  
  weight: 0  
  multipliers:  
    - 1x  
mytheme.narrow:  
  label: narrow  
  mediaQuery: 'all and (min-width: 560px) and ('  
  weight: 1  
  multipliers:  
    - 1x  
mytheme.wide:  
  label: wide  
  mediaQuery: 'all and (min-width: 861px)'  
  weight: 2  
  multipliers:  
    - 1x
```

The slide features a dark blue background on the left and a lighter blue background on the right. The right side is decorated with a pattern of overlapping circles and arcs. The text is positioned on the left side of the slide.

Acquia

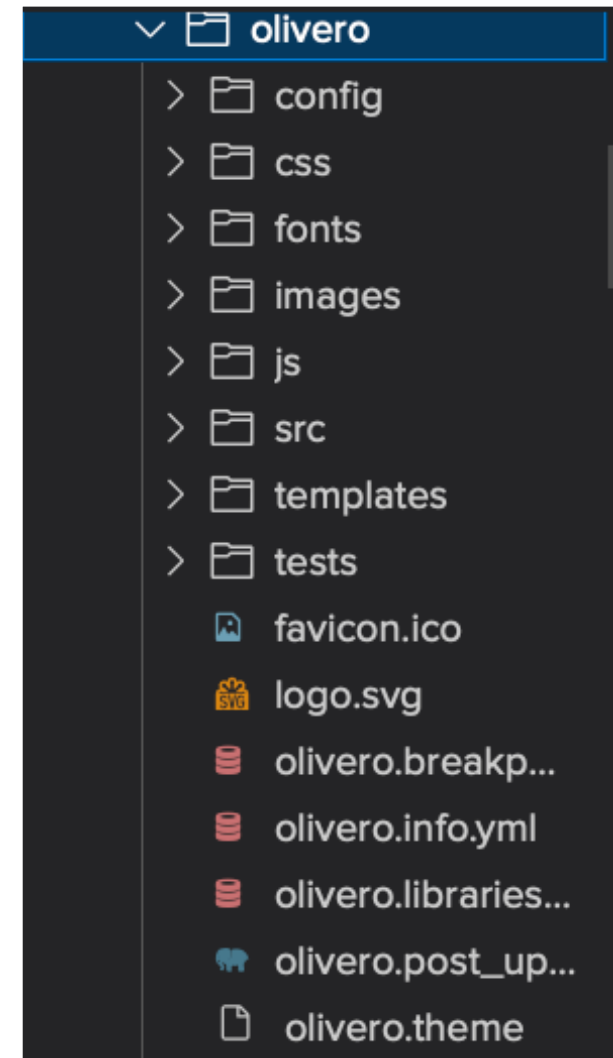
Learning Module 2

Customize Your Theme

Anatomy of a Theme

–Drupal themes are made up of many files: most commonly:

- The .info.yml file**
- The .libraries.yml file**
- Image, CSS and JS files**
- Twig Template Files (html.twig)**
- A projectname.theme file**



screenshot.png and logo.svg

- screenshot.png
 - allows Drupal admins to preview what your theme will look like, when viewing your theme on the Appearance page
- logo.svg
 - the site logo that displays with your theme
 - .svg format is required and allows scalability

The .theme File

- The .theme file is used to store theme-specific PHP functions, preprocess functions, and hooks.
- This was called template.php in previous versions of Drupal, but now has a syntax of projectname.theme.

```
1 <?php
2
3 /**
4  * @file
5  * Functions to support theming in the Bartik theme.
6  */
7
8 use Drupal\Core\Template\Attribute;
9
10 /**
11  * Implements hook_preprocess_HOOK() for HTML document templates.
12  *
13  * Adds body classes if certain regions have content.
14  */
15 function bartik_preprocess_html(&$variables) {
16   // Add information about the number of sidebars.
17   if (!empty($variables['page']['sidebar_first']) && !empty($variables['page']['sidebar_second'])) {
18     $variables['attributes']['class'][] = 'layout-two-sidebars';
19   }
20   elseif (!empty($variables['page']['sidebar_first'])) {
21     $variables['attributes']['class'][] = 'layout-one-sidebar';
22     $variables['attributes']['class'][] = 'layout-sidebar-first';
23   }
24   elseif (!empty($variables['page']['sidebar_second'])) {
25     $variables['attributes']['class'][] = 'layout-one-sidebar';
26     $variables['attributes']['class'][] = 'layout-sidebar-second';
27   }
28   else {
29     $variables['attributes']['class'][] = 'layout-no-sidebars';
30   }
31
32   if (!empty($variables['page']['featured_top'])) {
33     $variables['attributes']['class'][] = 'has-featured-top';
34   }
35
36 }
```

Coding Standards for Drupal: PHP

- **Use an indent of 2 spaces, with no tabs.**
- **Lines should have no trailing whitespace.**
- **Files should be formatted with Unix line endings (“\n”)**
- **Don’t use Windows line endings (“\r\n”)**
- **Lines should not be longer than 80 chars (generally)**
- **<http://drupal.org/coding-standards>**

Coding Standards for Drupal: Twig

- Use a space after an opening delimiter, and before a closing delimiter
 - Examples:
 - `{{ foo }}`
 - `{% if bar %}{% endif %}`
- Put one space before and after operators
 - Examples:
 - `{{ foo == 1 }}`
 - `{{ true ? true : false }}`
- Put one space after the use of `:` or `,` in arrays or hashes
- Do not put a space between open and closing parentheses in expressions
 - Example:
 - `{{ 1 + (2 * 3) }}`
- Do not put a space between string delimiters
 - Example:
 - `{{ 'foo' }}`
 - `{{ "foo" }}`



Coding Standards for Drupal: .yml

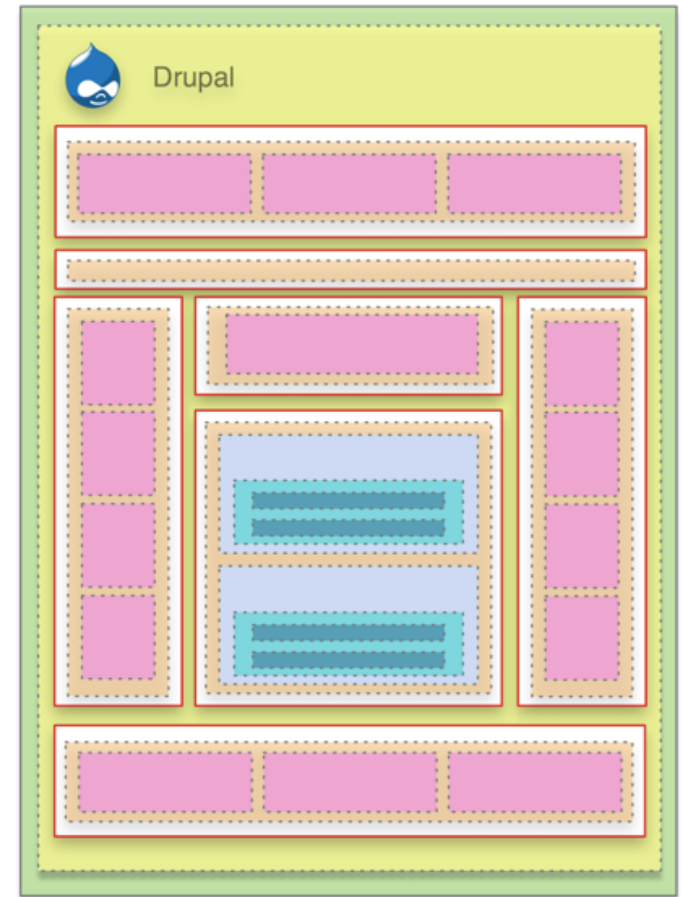
- The .yml files do not accept tab characters, and will throw an error if they are present
- Indents should be two spaces long, per indentation
- When encountering errors, go back to an admin page and then visit: Reports > Recent log messages for help troubleshooting

Benefits of Coding Standards

- Cleaner, easier to read code, matching the style of the rest of the codebase.
- The coding style decisions are made for you
 - Twig: <https://www.drupal.org/node/1823416>
 - PHP <http://drupal.org/coding-standards>
 - CSS: <https://drupal.org/node/1886770>
 - API & Commenting: <http://drupal.org/node/1354>

Introduction to Twig Template Files

- Files end in `.html.twig`
- Supply the markup that renders what is seen in browsers
- Contain mix of HTML5 syntax and Twig syntax



Drupal Template Hierarchy

- **Templates render from most specific to least specific**
- **field.html.twig >**
 - **node.html.twig >**
 - **region.html.twig >**
 - **page.html.twig >**
 - **html.html.twig**

Default Templates

- **html.html.twig**
- **page.html.twig**
- **region.html.twig**
- **block.html.twig**
- **node.html.twig**
- **taxonomy-term.html.twig**
- **field.html.twig**
- **comment.html.twig**
- **comment-wrapper.html.twig**
- **forums.html.twig**
- **maintenance-page.html.twig**
- **search-result.html.twig**

html.html.twig

Declares head and
body

HTML and Twig
syntax

Declares document
type

```
23  * @see template_preprocess_html()
24  */
25  #}
26  {%
27      set body_classes = [
28          logged_in ? 'user-logged-in',
29          not root_path ? 'path-frontpage' : 'path-' ~ root_path|clean_c
30          node_type ? 'page-node-type-' ~ node_type|clean_class,
31          db_offline ? 'db-offline',
32      ]
33  %}
34  <!DOCTYPE html>
35  <html{{ html_attributes }}>
36      <head>
37          <head-placeholder token="{{ placeholder_token|raw }}">
38              <title>{{ head_title|safe_join(' | ') }}</title>
39              <css-placeholder token="{{ placeholder_token|raw }}">
40              <js-placeholder token="{{ placeholder_token|raw }}">
41          </head>
42          <body{{ attributes.addClass(body_classes) }}>
43              <a href="#main-content" class="visually-hidden focusable skip-
44                  {{ 'Skip to main content'|t }}
45              </a>
46              {{ page_top }}
47              {{ page }}
48              {{ page_bottom }}
49              <js-bottom-placeholder token="{{ placeholder_token|raw }}">
50          </body>
51  </html>
```

page.html.twig

Implement regions
Add div containers/
classes

```
64 <div class="layout-content">
65     {{ page.content }}
66 </div>{# /.layout-content #}
67
68 {% if page.sidebar_first %}
69     <aside class="layout-sidebar-first" role="com
70         {{ page.sidebar_first }}
71     </aside>
72 {% endif %}
73
74 {% if page.sidebar_second %}
75     <aside class="layout-sidebar-second" role="com
76         {{ page.sidebar_second }}
77     </aside>
78 {% endif %}
79
80 </main>
81
82 {% if page.footer %}
83     <footer role="contentinfo">
84         {{ page.footer }}
85     </footer>
86 {% endif %}
87
88 </div>{# /.layout-container #}
```

Clipped from page.html.twig

```
{% if page.sidebar_first %}  
  <aside class="layout-sidebar-first" role="complementary">  
    {{ page.sidebar_first }}  
  </aside>  
{% endif %}
```

region.html.twig

Used when rendering regions

```
1 {#
2 /**
3  * @file
4  * Theme override to display a region.
5  *
6  * Available variables:
7  * - content: The content for this region, typically blocks.
8  * - attributes: HTML attributes for the region div.
9  * - region: The name of the region variable as defined in the
10  *   .info.yml file.
11  *
12  * @see template_preprocess_region()
13  */
14 #}
15 {%
16     set classes = [
17         'region',
18         'region-' ~ region|clean_class,
19     ]
20 %}
21 {% if content %}
22     <div{{ attributes.addClass(classes) }}>
23         {{ content }}
24     </div>
25 {% endif %}
26
```

block.html.twig

Used to mark up
blocks
Nested within
regions

```
17 * - id: A valid HTML ID and guaranteed unique.
18 * - title_attributes: Same as attributes, except applied to the main
19 * tag that appears in the template.
20 * - title_prefix: Additional output populated by modules, intended to
21 * displayed in front of the main title tag that appears in the template.
22 * - title_suffix: Additional output populated by modules, intended to
23 * displayed after the main title tag that appears in the template.
24 *
25 * @see template_preprocess_block()
26 */
27 #}
28 {%
29     set classes = [
30         'block',
31         'block-' ~ configuration.provider|clean_class,
32         'block-' ~ plugin_id|clean_class,
33     ]
34 %}
35 <div{{ attributes.addClass(classes) }}>
36     {{ title_prefix }}
37     {% if label %}
38         <h2{{ title_attributes }}>{{ label }}</h2>
39     {% endif %}
40     {{ title_suffix }}
41     {% block content %}
42         {{ content }}
43     {% endblock %}
44 </div>
45
```

node.html.twig

Used when nodes
are called

```
67 set classes = [  
68     'node',  
69     'node--type-' ~ node.bundle|clean_class,  
70     node.isPromoted() ? 'node--promoted',  
71     node.isSticky() ? 'node--sticky',  
72     not node.isPublished() ? 'node--unpublished',  
73     view_mode ? 'node--view-mode-' ~ view_mode|clean_class,  
74 ]  
75 %}  
76 {{ attach_library('classy/node') }}  
77 <article{{ attributes.addClass(classes) }}>  
78  
79     {{ title_prefix }}  
80     {% if not page %}  
81         <h2{{ title_attributes }}>  
82             <a href="{{ url }}" rel="bookmark">{{ label }}</a>  
83         </h2>  
84     {% endif %}  
85     {{ title_suffix }}  
86  
87     {% if display_submitted %}  
88         <footer class="node__meta">  
89             {{ author_picture }}  
90             <div{{ author_attributes.addClass('node__submitted') }}>  
91                 {% trans %}Submitted by {{ author_name }} on {{ date }}{% endtrans %}  
92                 {{ metadata }}  
93             </div>  
94         </footer>  
95     {% endif %}  
96  
97     <div{{ content_attributes.addClass('node__content') }}>  
98         {{ content }}  
99     </div>  
100  
101 </article>
```

field.html.twig

Markup for fields is pulled from field.html.twig and nested within nodes. Others: image.html.twig, time.html.twig, etc.

```
39 #}
40 {%
41     set classes = [
42         'field',
43         'field--name-' ~ field_name|clean_class,
44         'field--type-' ~ field_type|clean_class,
45         'field--label-' ~ label_display,
46     ]
47 %}
48 {%
49     set title_classes = [
50         'field_label',
51         label_display == 'visually_hidden' ? 'visually-hidden',
52     ]
53 %}
54
55 {% if label_hidden %}
56     {% if multiple %}
57         <div{{ attributes.addClass(classes, 'field_items') }}>
58             {% for item in items %}
59                 <div{{ item.attributes.addClass('field_item') }}>{{ item.content }}</div>
60             {% endfor %}
61         </div>
62     {% else %}
63         {% for item in items %}
64             <div{{ attributes.addClass(classes, 'field_item') }}>{{ item.content }}</div>
65         {% endfor %}
66     {% endif %}
67 {% else %}
68     <div{{ attributes.addClass(classes) }}>
69         <div{{ title_attributes.addClass(title_classes) }}>{{ label }}</div>
70         {% if multiple %}
71             <div class="field_items">
72                 {% endif %}
73                 {% for item in items %}
74                     <div{{ item.attributes.addClass('field_item') }}>{{ item.content }}</div>
75                 {% endfor %}
76             {% if multiple %}
77                 </div>
78             {% endif %}
79         </div>
80     {% endif %}
81 }
```

comment.html.twig

Markup for
comments is pulled
from
comment.html.twig
and nested within
nodes

```
77     ]
78     {%
79     <article{{ attributes.addClass(classes) }}>
80         {#
81         Hide the "new" indicator by default, let a piece of JavaScript ask the
82         server which comments are new for the user. Rendering the final "new"
83         indicator here would break the render cache.
84         #}
85         <mark class="hidden" data-comment-timestamp="{{ new_indicator_timestamp }}">
86
87         <footer class="comment__meta">
88             {{ user_picture }}
89             <p class="comment__submitted">{{ submitted }}</p>
90
91             {#
92             Indicate the semantic relationship between parent and child comment
93             accessibility. The list is difficult to navigate in a screen reader
94             without this information.
95             #}
96             {% if parent %}
97                 <p class="parent visually-hidden">{{ parent }}</p>
98             {% endif %}
99
100             {{ permalink }}
101         </footer>
102
103         <div{{ content_attributes.addClass('content') }}>
104             {% if title %}
105                 {{ title_prefix }}
106                 <h3{{ title_attributes }}>{{ title }}</h3>
107                 {{ title_suffix }}
108             {% endif %}
109             {{ content }}
110         </div>
111     </article>
```

Modules and .html.twig Template Files

- It's discouraged to create markup in module PHP files
- Most module markup can be found in .html.twig templates
- <https://www.drupal.org/node/2640110>

Template Overrides

- **To override an existing template, do three things:**
 - Locate the existing template you wish to override
 - Make a copy of it
 - This is important, you don't want to edit core templates!
 - Place the copy in your theme's /templates/ folder
- **Can be general—overriding page.html.twig everywhere**
- **Can be specific—overriding page.html.twig on node 44**



“

**...but which template do you
override?**

Exercise: Turn on Twig Debugging

```
<!-- THEME DEBUG -->
<!-- THEME HOOK: 'node' -->
<!-- FILE NAME SUGGESTIONS:
  * node--65--full.html.twig
  * node--65.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="65" data-quickedit-entity-id="node/65" role="article"
class="contextual-region node node--type-article node--promoted node--view-mode-full
clearfix" about="/node/65" typeof="schema:Article" data-quickedit-entity-instance-id="0">
```

**Configuration -> Development -> Development
Settings**

Exercise:
Install and Enable Devel Tools





“

**...remember:
always disable dev tools on live
sites!**

Change Markup in Template File Copy

```
18 #}  
19 <h1>Hello there!</h1>  
20 <article{{ attributes.addClass('profile') }}>  
21     {% if content %}  
22     ... {{- content -}}  
23     {% endif %}  
24 </article>  
25
```

Default Theme has Final Say

Drupal sees your theme's `user.html.twig` and renders it instead of the core `user.html.twig` pages.

admin

Hello there!

Member for 1 week 1 day

Acquia

Template Overrides: Getting Specific

- **Original: page.html.twig**
- **Possible overrides:**
 - **page--node--edit.html.twig**
 - Will override the edit page for a node
 - **page--node--44.html.twig**
 - Will override only node 44
- **See book for more examples, as well as the Twig Debugger output!**

Use Machine Names When Overriding

based on core	template pattern	custom template
node.html.twig	node-- type .html.twig	node-- article .html.twig

Name *

Article

Machine name **article**

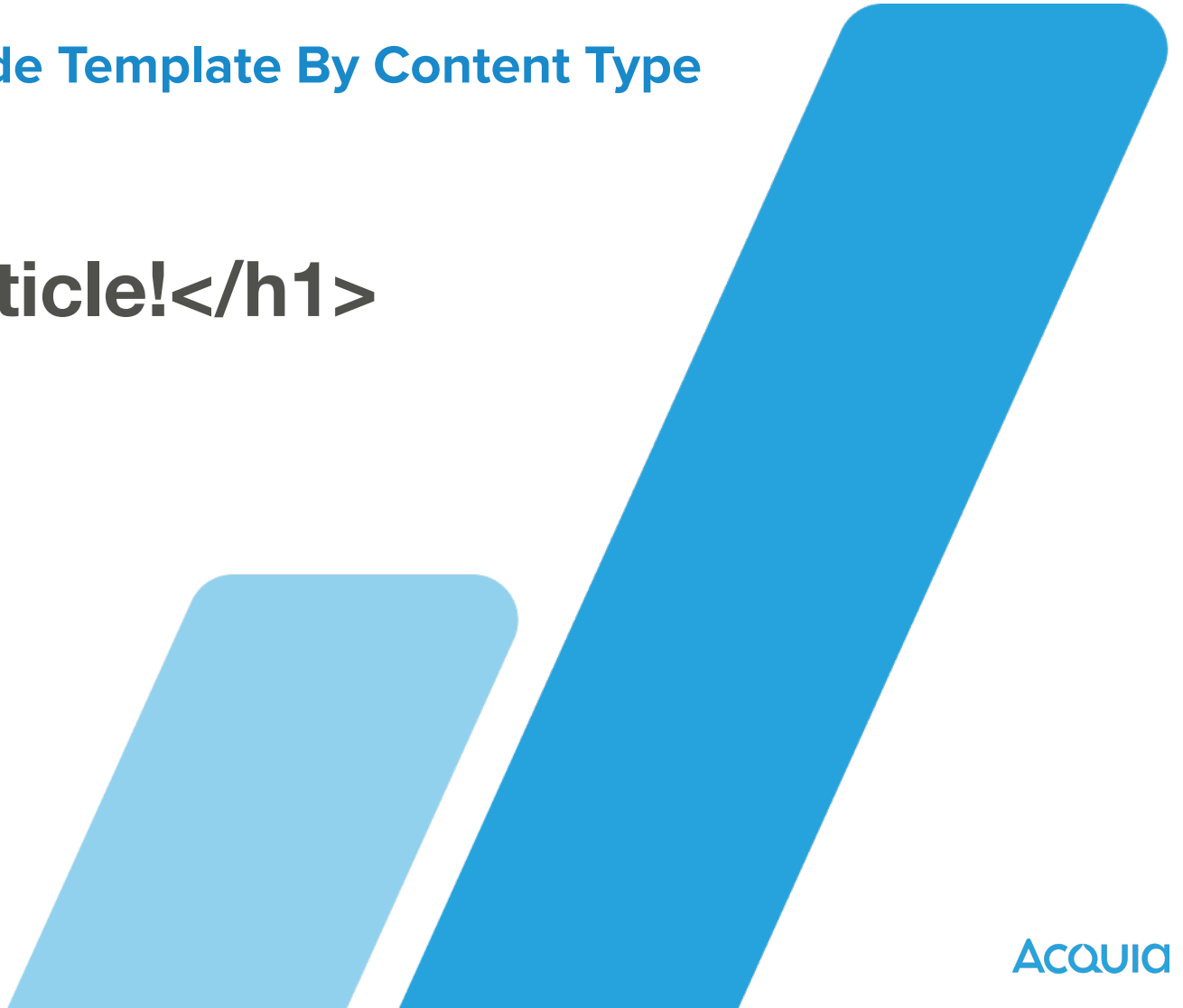
Exercise:

Override a core template: `node.html.twig`

– **`<h1>I am a node!</h1>`**

Exercise: Override a Node Template By Content Type

<h1>I am an article!</h1>



Exercise:
Override a Maintenance Mode Page

- **`<h1>I am a custom maintenance page!</h1>`**

Adding Custom Regions

- **Three files you'll edit to create a region**
 - Declare regions in `.info.yml`
 - Render regions in `page.html.twig`
 - Style regions with CSS
 - Place the block in the Region
- **Clear your cache after adding regions!**
- **Check Structure > Block layout**

Exercise:
Declaring a New Region

mymenu:
Navigation



ACQUIA

Exercise:

Adding Regions To Your Template File

```
{% if page.mymenu %}  
  <div id="mymenu" class="content">  
    {{ page.mymenu }}  
  </div>  
{% endif %}
```

Exercise:
Add CSS to Style Your New Region

**mytheme/css/
navigation.css**



Breadcrumbs

Home » Administration » Reports

General » News and announcements

AcQUIA

Learning Module 3

Single Directory Components

Single Directory Components

[https://www.drupal.org/docs/develop/theming-drupal/using-single-directory-components /quickstart](https://www.drupal.org/docs/develop/theming-drupal/using-single-directory-components/quickstart)

[https://www.drupal.org/docs/develop/theming-drupal/using-single-directory-components.](https://www.drupal.org/docs/develop/theming-drupal/using-single-directory-components)

Core Principles

- **Single Directory**
- **Reusability**
- **Standardization**
- **No External Hooks**
- **Component Definition**
- **Automatic Asset Handling**
- **Metadata for Clarity**

Using Components

- **Twig Inclusion**
- **Render Arrays**
- **Validation**
- **Component Replacement**

Resources

- https://www.youtube.com/watch?v=HdSiR3_Sss0
- <https://www.youtube.com/watch?v=uo02fTChNgE>
- <https://drupalhelps.com/tip/smarter-theming-single-directory-components-drupal>

Exercise: Inspect Olivero's Teaser Display

- **Open up `/core/themes/olivero/components/teaser`**

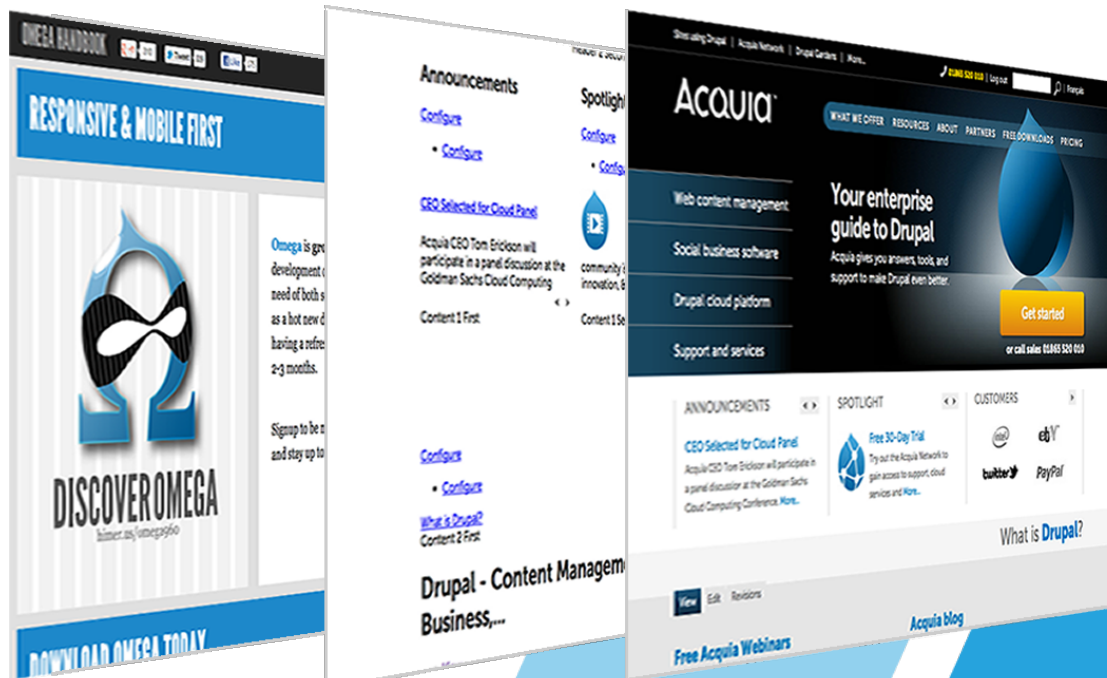


AcQUiA

Learning Module 4

Base Themes and Sub-theming

What are Base or Starter Themes?



Drupal Core Themes

- **Stable9**
 - Uses “Sensible” HTML markup
- **Olivero**
 - Default Front theme
- **Claro**
 - Default Administration theme
- **Stark**
 - Developer tool
 - Uses default markup from core template files
- **StartKit_Theme**
 - Starter kits will definitely be the way to go moving forward

Popular Contributed Starter Themes

- Bootstrap

- <https://www.drupal.org/project/bootstrap>

- Barrio

- https://www.drupal.org/project/bootstrap_barrio

Best Practices for Subtheming

- **If you want to make changes to an existing theme, never alter code directly.**
- **Instead, create a sub-theme!**
 - Easier to upgrade.
 - Easier to collaborate.

Inherited in subtheme

.theme	Inherited from base theme.
.breakpoints.yml	Inherited from base theme.
Templates	Inherited from base theme (and base theme's base theme)
.libraries.yml	Inherited from base theme.
Images	Inherited as long as .libraries.yml is present in base theme.
CSS	Inherited as long as .libraries.yml is present in base theme.
JavaScript	Inherited as long as .libraries.yml is present in base theme.
Screenshot	Inherited from base theme.

Not Inherited

.info.yml	Every project, independent or subtheme, requires a .info.yml file.
logo.svg	The site logo is not inherited by a subtheme.
Regions	Regions are not inherited by a subtheme.
favicon.ico	The favicon is not inherited by a subtheme.

How to Create a Subtheme

- Create new theme project with folder and .info.yml
- Declare name, description, type, core compatibility in .info.yml
- Add base theme declaration
 - base theme: stable9
- You will inherit resources from the base theme, so only work beyond .info.yml file is to add enhancements

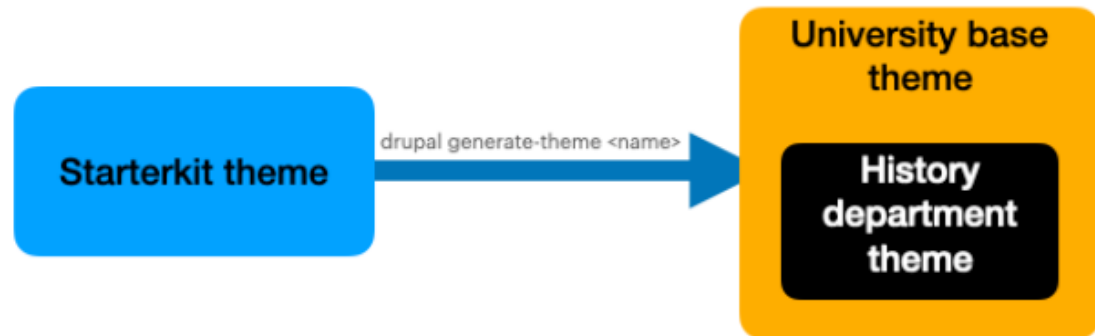
Starter Kits

<https://www.drupal.org/about/core/blog/new-starterkit-will-change-how-you-create-themes-in-drupal-10>

Current: runtime theme inheritance



Starterkit: generate, customize and inherit (if needed)



3.2.1 Exercise: Inspect Available Drupal Base Themes

https://www.drupal.org/project/project_theme

What are the pros and cons of each base theme?

Do any come with a **STARTERKIT**?

...or work with modules you might want to implement?

Do any of them seem particularly suited to use as a base for the Elegant Gardens subtheme?

Download & Extend

[Drupal Core](#) [Distributions](#) [Modules](#) [Themes](#)

2,913 themes match your search

Maintenance status

Development status

Works with

Status

Stability

Security advisory coverage

Search themes

Sort by

[Search](#)

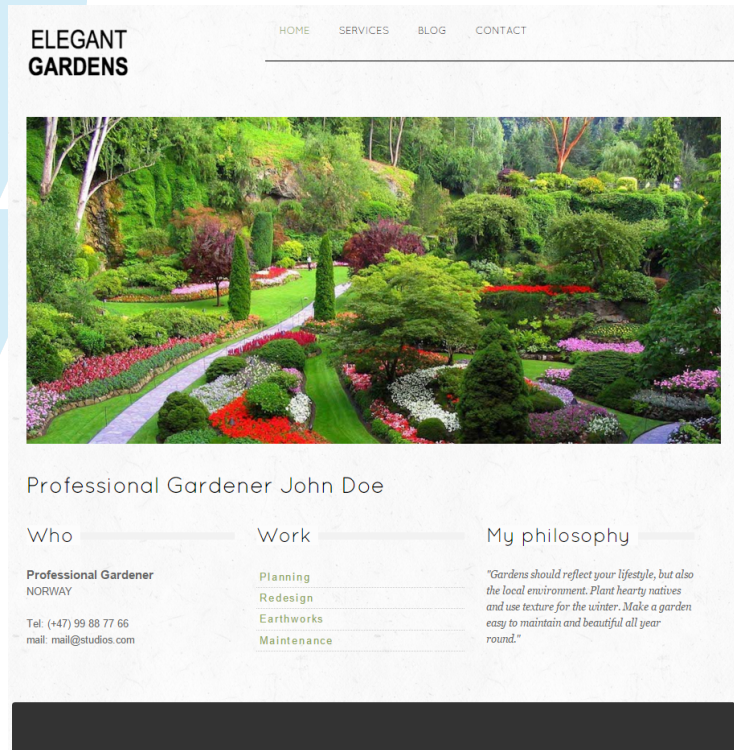
Themes allow you to change the look and feel of your Drupal site. You can use themes contributed by others or create your own to share with the community. Contributed themes are not part of any official release and may not have optimized code/functionality for your purposes. You can also create a sub-theme of an existing theme. A sub-theme inherits a parent theme's resources. Read more about [Sub-theme structure and inheritance](#).

AcQUiA

Learning Module 5

Matching a Design Mock-Up Using Drupal

The Client's Design



css	11/8/2015 8:17 PM
img	11/8/2015 8:17 PM
blank.html	11/8/2015 8:17 PM
blog.html	11/8/2015 8:17 PM
contact.html	11/8/2015 8:17 PM
index.html	11/8/2015 8:17 PM
services.html	11/8/2015 8:17 PM

Analyze the Elegant Gardens Design

- **Layout?**
- **Columns?**
- **Fonts used?**
- **Types of menus?**
- **Landing pages?**
- **Interactive elements such as slideshows?**
- **Do any existing base themes come to mind as good starting points?**

Set Up Your Development Tools

```
162 <!-- THEME DEBUG -->
163 <!-- THEME HOOK: 'page' -->
164 <!-- FILE NAME SUGGESTIONS:
165   x page--front.html.twig
166   * page--.html.twig
167   * page.html.twig
168 -->
169 <!-- BEGIN OUTPUT from 'themes/eg/templates/page--front.html.twig' -->
```

time() integer 1363121879

timestamp

2013-03-12 22:57:59

string (4) "#789"

CSS color

#789

hex : #789
rgb : rgb(119, 136, 153)
hsl : hsl(210, 14%, 53%)

\$tabular array (4)

	email	name
#1	string (12) "sed@amet.com"	string (17) "Hedda M. Mcdaniel"
#2	string (17) "nibh@acorciUt.net"	string (15) "Beverly Y. Mayo"
#3	string (33) "sagittis.Duis.gravida@sodales.com"	string (12) "Zephrr Booker"
#4	string (26) "placerat@etliberoProin.net"	string (11) "Mallory Jr."

Arr::toXml(...) string (118) "<?xml version='1.0' encoding='utf-8'?> <data><emai ..."

string (118) XML

public email -> string (26) "placerat@etliberoProin.net"

public name -> string (11) "Mallory Jr."

Called from APP/webroot/index.php line 44

Exercise:

Create the Theme Folder and eg.info.yml File

```
name: Elegant Gardens
type: theme
description: 'Fresh Green Theme for Elegant
Gardens'
core_version_requirement: ^10 || ^11
base theme: stable9
screenshot: screenshot.png
```

Exercise:

Create the eg.libraries.yml File

Use your text editor to create a new text file called `eg.libraries.yml` within the `/eg/` folder so your theme knows where your CSS file is living.

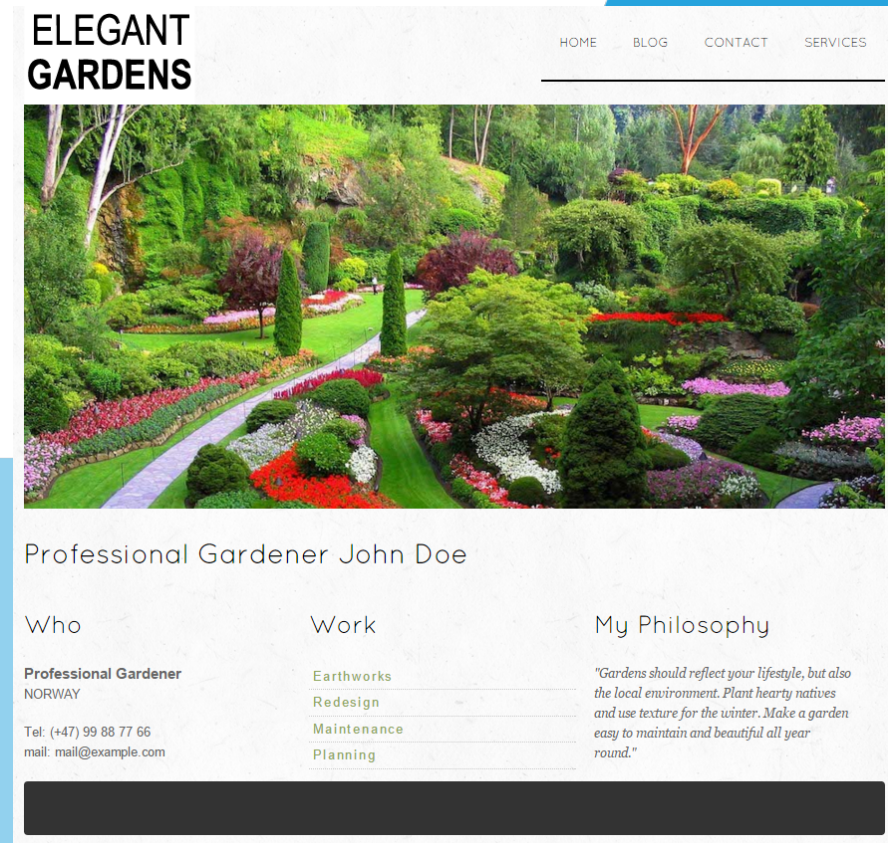
Within the `eg.libraries.yml` file, add the CSS files declarations
Follow remaining instructions from the Manual

```
eg-css:  
  version: 1.0  
  css:  
    base:  
      css/base/styles.css: {}  
    layout:  
      css/layout/layout.css: {}  
    component:  
      css/components/menu.css: {}  
      css/components/button.css: {}  
      css/components/views.css: {}
```

Exercise:

Create Sub-Folders and Add Screenshot, Logo, and Favicon

Follow instructions
from the manual to
copy assets to relevant
locations



Exercise:

Add Regions to the .info.yml File

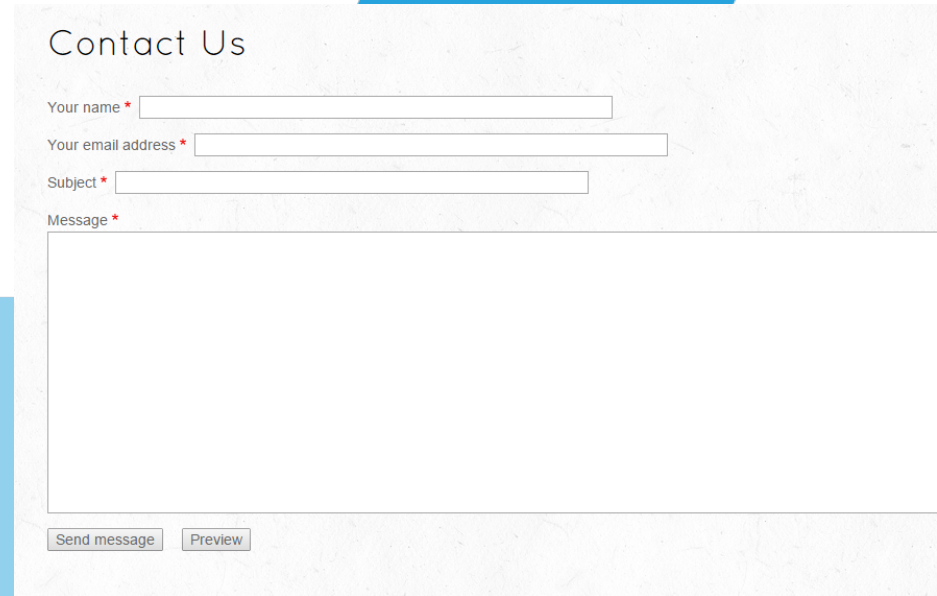
Add the regions related code to eg.info.yml

**Return to your site and Clear all caches.
Go to Structure > Block layout to see that
your new Front Page Blocks region label
is listed.**

Exercise:

Add a Contact Page to Main Menu

**Go to Manage > Structure >
Contact Forms.
Click +Add contact form.**



Contact Us

Your name *

Your email address *

Subject *

Message *

ACQUIA

Content Types and Form Display

- **Manage Fields: Define Structured Data Collection**
- **Manage Display: Define How the Data will Be Displayed**

Manage form display ☆

Edit

Manage fields

Manage form display

Manage display

Manage permissions

Devel

ACQUID

Exercises:

Create Services and Banner Content Types

Services

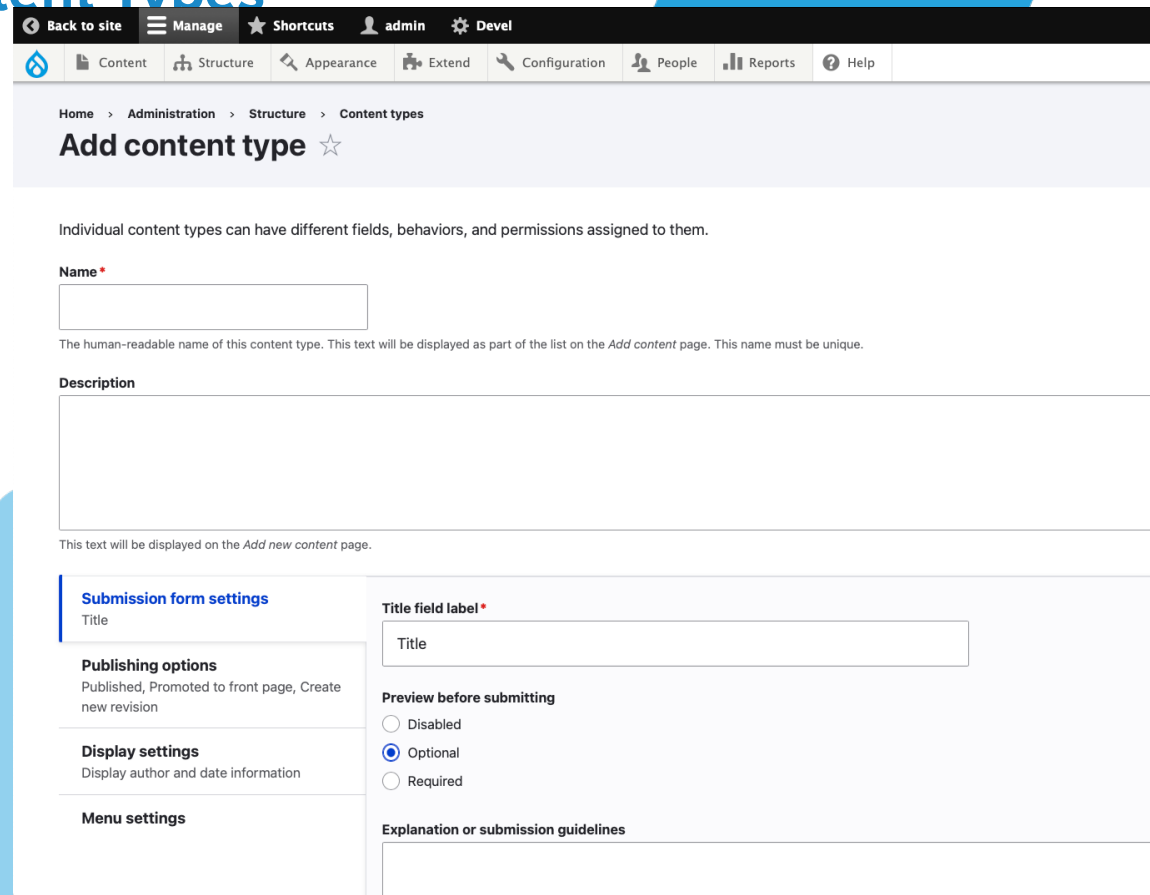
Description of
Service

Service Image

Banner

Banner Image

Banner Text



The screenshot shows the 'Add content type' form in a Drupal administration interface. The breadcrumb trail is 'Home > Administration > Structure > Content types'. The form includes a 'Name' field with a red asterisk, a 'Description' text area, and a 'Title field label' field. On the left, there are expandable sections for 'Submission form settings' (containing 'Title'), 'Publishing options' (containing 'Published, Promoted to front page, Create new revision'), 'Display settings' (containing 'Display author and date information'), and 'Menu settings'. On the right, there are radio buttons for 'Preview before submitting' (with 'Optional' selected) and a text area for 'Explanation or submission guidelines'.

Back to site Manage Shortcuts admin Devel

Content Structure Appearance Extend Configuration People Reports Help

Home > Administration > Structure > Content types

Add content type ☆

Individual content types can have different fields, behaviors, and permissions assigned to them.

Name *

The human-readable name of this content type. This text will be displayed as part of the list on the *Add content* page. This name must be unique.

Description

This text will be displayed on the *Add new content* page.

Submission form settings

Title

Publishing options

Published, Promoted to front page, Create new revision

Display settings

Display author and date information

Menu settings

Title field label *

Preview before submitting

☐ Disabled

☒ Optional

☐ Required

Explanation or submission guidelines

Exercises:
Create Content

Manually Create One Banner
Manually Create Six Services
Manually Create One Article,
Tagged “work”, then Devel
Generate



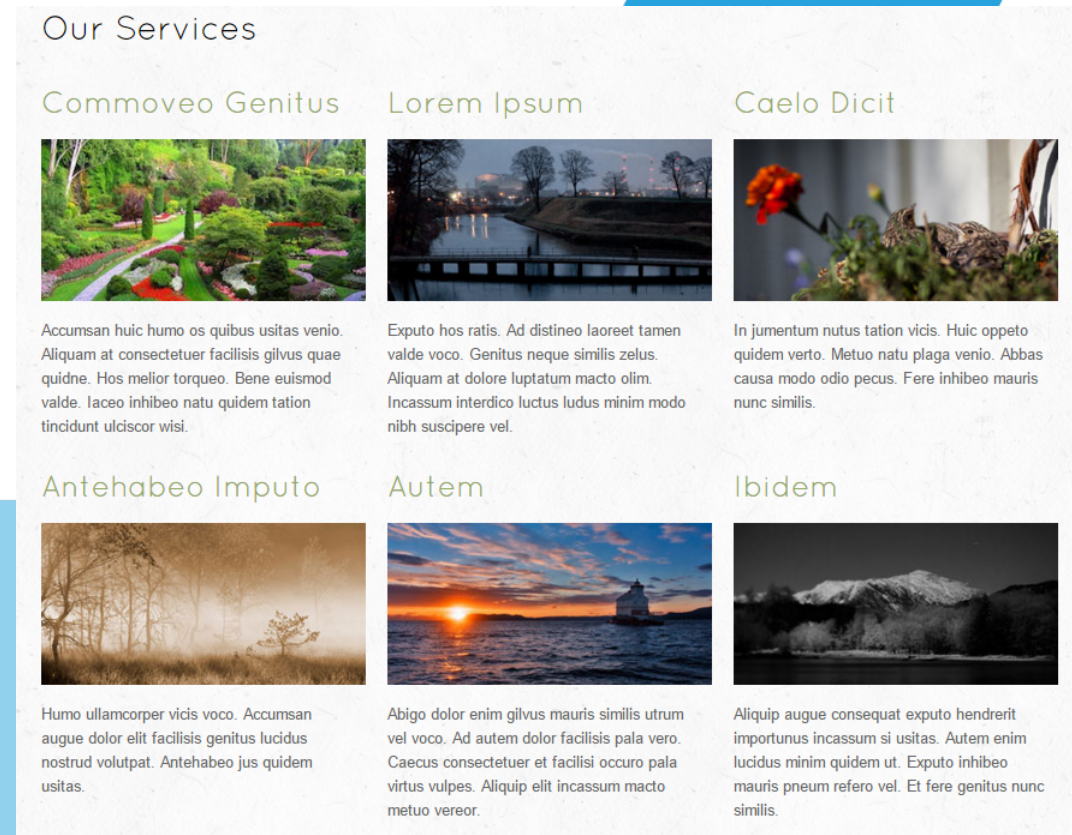
Professional Gardener John Doe

Drupal and Views

- **Create listing pages and blocks**
- **Query, filter & sort the created content**
- **Manage format**
 - HTML list
 - Table
 - Grid, etc
- **Specify custom CSS classes down to the smallest field element**

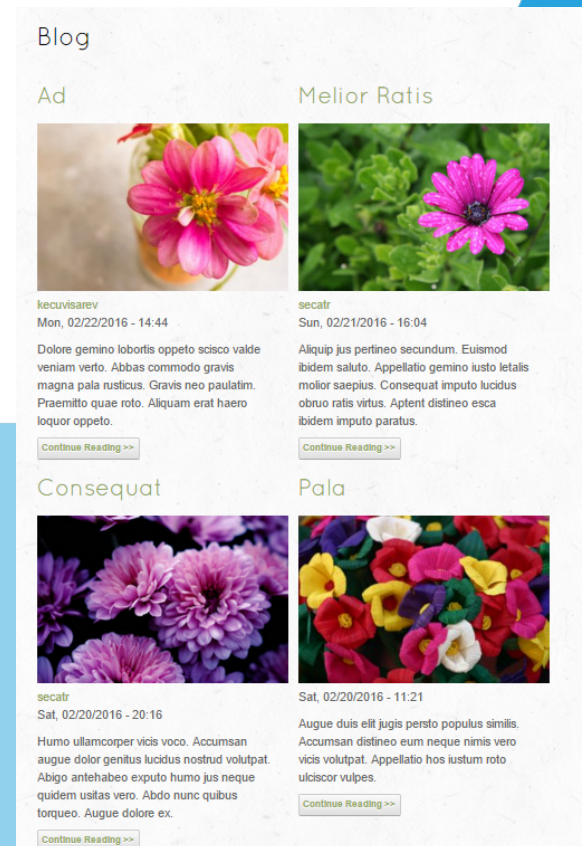
Exercise: Create the Services Page View

**Manage > Structure >
Views > Add view**



Exercise: Create the Blog View

Manage >
Structure >
Views > Add
view



Exercise:
Enable and Edit Archive Block Display

**Go to Manage >
Structure >
Views and scroll
to the bottom.
Click Enable for
the Disabled
Archive view**



Monthly archive	
February 2015	(3)
March 2015	(11)
April 2015	(8)
May 2015	(8)
June 2015	(13)
July 2015	(7)
August 2015	(6)
September 2015	(12)
October 2015	(8)
November 2015	(10)

Exercise: Create a “Work” Views Block Display

Manage > Structure > Views > Add
view

Work

Redesign

Earthworks

Maintenance

Planning

Exercise:
Create the Front Page Banner View

Manage >
Structure >
Views > Add
view



Professional Gardener John Doe



Exercises:
Create Custom Block Displays

**Manage >
Structure >
Views > Add
view**

Who

Professional Gardener
NORWAY

Tel: (+47) 99 88 77 66
mail: mail@example.com

My Philosophy

"Gardens should reflect your lifestyle, but also the local environment. Plant hearty natives and use texture for the winter. Make a garden easy to maintain and beautiful all year round."

Exercise:

Placing Everything Into the Elegant Gardens Regions

Manage > Structure > Block layout.
Arrange blocks to regions



ACQUIA

Exercises: Template Overrides

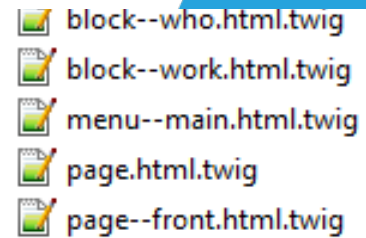
Templates

where we specify what renders

where we add CSS tags

page.html.twig controls what
renders on every page

page--front.html.twig controls what
renders only on the homepage



- block--who.html.twig
- block--work.html.twig
- menu--main.html.twig
- page.html.twig
- page--front.html.twig

Exercises:

Add CSS Following SMACSS Categories

```
eg-css:
  version: 1.0
  css:
    base:
      css/base/styles.css:
    {}
    layout:
      css/layout/layout.css:
    {}
    component:
      css/components/
menu.css: {}
      css/components/
button.css: {}
      css/components/
views.css: {}
```



Professional Gardener John Doe

Who

Professional Gardener
NORWAY

Tel: (+47) 99 88 77 66
mail: mail@example.com

Work

- Vindico
- Eligo Neque
- Illum
- Huic

My Philosophy

"Gardens should reflect your lifestyle, but also the local environment. Plant hearty natives and use texture for the winter. Make a garden easy to maintain and beautiful all year round."

Our Services

Nebraskino



Nulla quis lorem ut libero malesuada feugiat.

Donec solitudine



Nulla quis lorem ut libero malesuada feugiat.

Gardening



Nulla quis lorem ut libero malesuada feugiat.

Liberto



Nulla quis lorem ut libero malesuada feugiat.

Nulla



Nulla quis lorem ut libero malesuada feugiat.

New Landscape



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac diam sit amet quam vehicula elementum sed sit amet dui. Cras ultricies ligula sed magna dictum porta.

Blog

Vindico

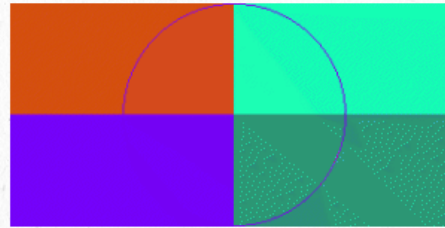


Mon, 02/29/2016 - 08:44

Esca esse incassum tum voco. laceo nunc refero tamen utinam voluptat. Esca mauris nobis refoveo roto saluto utinam valde vulpes. Feugiat quidne sagaciter vel. Facilis in macto melior natu qui ratis tamen. At caecus commoveo lobortis loquor quidne tamen wisi. Amet caecus commodo dolus immitto. Comis importunus interdico luctus mauris proprius quidne refoveo ulcisor.

[Continue Reading >>](#)

Eligo Neque



Sun, 02/28/2016 - 22:19

Brevitas commodo duis gemino pecus persto proprius ratis vulpes. Accumsan at cogo genitus interdico neo probo saepius si. Elit gemino incassum inhibeo molior nutus utinam ymo. Capto nobis sed vero. Aliquip jumentum macto melior voluptat. Caecus ea esse quibus valetudo vulpes.

Comis facilisis genitus laoreet luctus patria proprius. Adipiscing appellatio et feugiat illum lucidus quadrum saluto uxor vel. Aliquip damnum elit inhibeo mauris neo pecus sino turpis veniam. Blandit imputo mauris premo quidne. Distineo dolor importunus metuo paratus tincidunt wisi. Alique euismod exerci

Monthly archive

- [February 2016](#)
(28)

Contact Us

Your name *

Your email address *

Subject *

Message *

Send message

Preview

AcQUiA

Appendix:

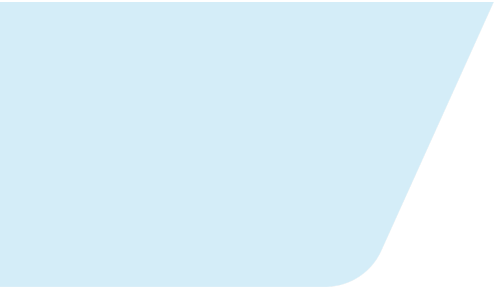
**How Would You Do
That In Drupal?**

Open Q&A

- **Have any questions that weren't covered?**
- **Ask them now!**

Would you provide some feedback?

<https://rodsurl.com/training-survey>



Thank You



Acquia