



# Module 2: Branding the Platform with Themes

## Learning Objectives

In this module, you'll learn how to control the branding of the platform by creating and customizing theme modules.

## Tasks to Accomplish

- Use the NPM Theme generator to generate a custom theme
  - Customize the HTML, CSS, and JS source files
  - Create custom platform settings in the theme
- Create Stylebooks to modify content pages on top of the Theme

## Exercise Prerequisites

- Java JDK installed to run Liferay
  - Download here: <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
  - Instructions on installation here: [https://www.java.com/en/download/help/download\\_options.xml](https://www.java.com/en/download/help/download_options.xml)
- Liferay Tomcat bundle installed in the following folder:
  - Windows: `C:\liferay\bundles`
    - Start Liferay by double-clicking the `startup.bat` file.
  - Unix Systems: `[user-home]/liferay/bundles`
    - Start Liferay by using `catalina.sh` run in the Terminal.
- Unzipped module exercise files in the following folder structure:
  - Windows: `C:\liferay`
  - Unix Systems: `[user-home]/liferay`
- NPM, Yeoman, and the Liferay Theme Generator installed
  - Download Node here: <https://nodejs.org/en/>
    - Some older Node versions may not work with the Liferay theme generator on DXP 7.4
  - Run `npm install -g yo`
  - Run `npm install -g generator-liferay-theme`
    - **Note:** OSX and Linux Users may need to redirect the NPM global command. See module 1 exercise instructions for more details.

- Liferay DXP or CE 7.4 instance already setup
  - Platform Name: Livingstone Hotels & Resorts
  - Admin User: Josiah Copeland
  - Admin Email: [josiah.copeland@livingstone.com](mailto:josiah.copeland@livingstone.com)

# Table of Contents

- Create the Livingstone Fjord Theme
- Modify the Styling of the Themes
- Add Custom JavaScript to a Theme
- Configure the Theme to Provide Administrative Options
- Adding Style Books
- Create Site Branding Module Quiz
- Answer Key

# Create the Livingstone Fjord Theme

## Exercise Goals

- Install Visual Studio Code and add code snippets
- Use the NPM Theme Generator to create a new theme
  - Set up some initial theme variables
  - Modify the HTML source file and add Footer code
- Deploy the Initial Base theme

## Install Visual Studio Code

1. **Go to** <https://code.visualstudio.com/>.
2. **Click** to download the VS Code for your OS.
3. **Double-click** to run the installer for your operating system.
4. **Click** through the installation steps.

## Create a Global Snippets File in Visual Studio Code

1. **Open** *Visual Studio Code*.
2. **Go to** the *User Snippets* menu.
  - Windows: File > Preferences > User Snippets
  - OSX: Gear Icon > User Snippets
3. **Click** on *New Global Snippets file...* in the drop-down menu.
4. **Type** *lfr-front-end* for the *name*.
  - The full file name will be *lfr-front-end.code-snippets*.
5. **Click** *Save*.

## Add the Snippet Content

1. **Open** the `lfr-front-end-snippets.json` in your module exercises folder.
2. **Copy** the contents of the file.
3. **Paste** the contents of the file in the `lfr-front-end.code-snippets` file just created in Visual Studio Code.
  - Make sure to replace all the existing content in the file.
4. **Save** the file.

## Start the Liferay-Tomcat Bundle

1. **Go to** the Tomcat server's `bin` directory:
  - Windows: `C:\liferay\bundles\liferay-dxp-[version]\tomcat-[version]\bin` in the file manager
  - Mac/Linux: `[userhome]/liferay/bundles/liferay-dxp-[version]/tomcat-[version]/bin` using the *Terminal*
2. **Start** the Tomcat Server:
  - Windows: **Double-click** on the `startup.bat`.
  - Mac/Linux: **Run** `./catalina.sh run` in your *Terminal*.

## Generate the Base Theme

1. **Go to** the *liferay* folder you already created in the *Command Line/Terminal*.
  - Windows: *C:\liferay*
  - Mac/Linux: *~/liferay*
2. **Run** the `yo liferay-theme` command.
  - This command will create the base theme files and install the necessary dependencies for deployment.
3. **Type** `Livingstone Fjord Theme` to name your theme.
4. **Press** *Enter*.
5. **Press** *Enter* to accept the default *themeld*.
  - This is how Liferay refers to your theme internally.
6. **Choose** `7.4` for the version.
7. **Press** *Enter*.
8. **Type** `N` when asked to use *Font Awesome* in the theme.
  - Font Awesome is no longer included by default and using it may cause errors.
9. **Press** *Enter*.
10. **Choose** *Local App Server* when asked to select your deployment strategy.
11. **Type** the directory:
  - Windows: *C:\liferay\bundles\liferay-dxp-[version]\tomcat-[version]*
  - Mac/Unix: */Users/[username]/liferay/bundles/liferay-dxp-[version]/tomcat-[version]*
12. **Press** *Enter*.
13. **Press** *Enter* again to use *http://localhost:8080* for the URL.

*Note: If the Theme Generator returns an error, make sure you are using Node.js 16.13.2 or another compatible version. Additionally, some setups run into issues running scripts with npm. If this is the case, try creating an .npmrc file in the root of your project and add the `ignore-scripts=false` configuration.*



## Deploy the Base Theme to Our Liferay Instance

1. **Go to *Livingstone Fjord Theme*** using `cd livingstone-fjord-theme` in the *Command Line*.
  - This is assuming you are still running *Command Prompt/Terminal* from `C:\liferay`.
2. **Run the `npm run deploy` command.**
  - This makes the base theme available in our Liferay instance.
3. **Open** your browser.
4. **Go to** `localhost:8080` if you're not already there.
5. **Sign in** to the *Livingstone Hotels & Resorts* platform.
6. **Open** the *Site Administration Panel*.
7. **Go to** *Site Builder > Pages* in *Site Administration* for the *Livingstone Hotels & Resorts* Site.
8. **Open** the *Configuration menu* next to *Public Pages*.
  - This is the gear icon.
9. **Click** the *Change Current Theme* button near the bottom.
10. **Choose** *Livingstone Fjord Theme*.
11. **Click** the *Save* button at the bottom.
12. **Click** *Home* in *Site Administration* to see your theme.

## Add the Directory Structure to the SRC Folder

1. **Go to** the `livingstone-fjord-theme\src` folder.
  - Windows: `C:\liferay\livingstone-fjord-theme\src`
  - Mac/Linux: `~/liferay/livingstone-fjord-theme\src`
2. **Create** the following folders:
  - `js`
  - `templates`

*Note: We will be using `npm run deploy` throughout the course. If you decide to use `gulp watch`, you will not need to run `npm run deploy`.*

## Set Variables to Start

1. **Copy** the `liferay-look-and-feel.xml` file from the `WEB-INF` folder in your `exercise-src` folder.
2. **Paste** the file into the `livingstone-fjord-theme\src\WEB-INF` folder.
  - You will need to replace the original file.

```
<?xml version="1.0"?>

<!DOCTYPE look-and-feel PUBLIC "-//Liferay//DTD Look and Feel 7.4.0//EN" "http://www.liferay.com/dtd/liferay/7.4.0/look-and-feel/look-and-feel.dtd">

<look-and-feel>

    <compatibility>

        <version>7.4.0+</version>

    </compatibility>

    <theme id="livingstone-fjord-theme" name="Livingstone Fjord Theme">

        <template-extension>ftl</template-extension>

        <settings>

            <setting configurable="false" key="color-palette" value="primary"></setting>
            <setting configurable="true" key="show-footer" type="checkbox" value="true"></setting>
            <setting configurable="true" key="show-header" type="checkbox" value="true"></setting>
            <setting configurable="true" key="show-main-navigation-in-full-screen"></setting>
            <setting configurable="false" key="show-site-name-default" value="true"></setting>
            <setting configurable="false" key="show-site-name-supported" value="true"></setting>
            <setting configurable="true" key="use-a-retina-logo" type="checkbox" value="true"></setting>

            <!-- Insert snippet 02-theme-settings here -->

        </settings>

        <!-- Insert snippet 01-portlet-decorators here -->

    </theme>

</look-and-feel>
```

## Modify Theme Templates

1. **Copy** the files in the `exercise-src/templates` folder.
2. **Paste** the contents into the `livingstone-fjord-theme|src|templates` folder.
3. **Open** the *Visual Studio Code* editor.
4. **Drop** the `portal_normal.ftl` file from the `livingstone-fjord-theme|src|templates` file into the *Visual Studio Code* editor.

## Modify the Header

1. **Click** to highlight the `<!-- Insert snippet 01-portal-normal-header here -->` comment.
2. **Type** `lfr` to view the available code snippets.
3. **Choose** the `ftl: 01-portal-normal-header` snippet.
  - Alternatively, you can just type in the following:

```
<#if show_header>

    <#include "${full_templates_path}/header.ftl" />

</#if>
```

## Add the Content Section

1. **Click** to highlight the `<!-- Insert snippet 02-portal-normal-main here -->` comment.
2. **Type** `lfr` to view the available code snippets.
3. **Choose** the `ftl: 02-portal-normal-main` snippet.

## Keep the Footer Code Distinct

1. **Click** to highlight the `<!-- Type footer.ftl include here -->` comment.
2. **Type** in the following to include the `footer.ftl` file and then *Save* the file:

```
<#if show_footer>

    <#include "${full_templates_path}/footer.ftl" />

</#if>
```

## Add Code to the Footer FTL File

1. **Drop** the `footer.ftl` file from `livingstone-fjord-theme\src\templates` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert 03-footer-ftl here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `03-footer-ftl` snippet.
5. **Save** the file.

## Add a Sign-In Link to the Footer

1. **Drop** the `footer_navigation.ftl` file from `livingstone-fjord-theme\src\templates` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 04-footer-nav-sign-in here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `04-footer-nav-sign-in` snippet.
  - Alternatively, you can type the following:

```
<#if !is_signed_in>

    <li class="nav-item" role="presentation">

        <a class="nav-link" data-redirect="${is_login_redirect_required?
string}" href="${sign_in_url}" id="sign-in" rel="nofollow">

            ${sign_in_text}

        </a>

    </li>

</#if>
```

## Add the Footer Navigation

1. **Click** to highlight the `<!-- Insert snippet 05-footer-nav-menu here -->` comment.
2. **Type** `lfr` to view the available code snippets.
3. **Choose** the `05-footer-nav-menu` snippet.
4. **Save** the file.
  - Alternatively, you can type and save the following:

```
<#foreach nav_item in nav_items>

    <li class="nav-item" role="presentation">

        <a class="nav-link" aria-labelledby="layout_${nav_item.getLayoutId()}"
        href="${nav_item.getURL()}" ${nav_item.getTarget()} role="menuitem">

            ${nav_item.getName()}

        </a>

    </li>

</#foreach>
```

## Deploy the Theme to See the HTML Changes

1. **Run** `npm run deploy` in the *Command Line* or *Terminal*.
2. **Open** your browser.
  - You should already be at `localhost:8080` and logged into your Liferay instance.
3. **Click** to *Refresh* the page after the changes to your theme have finished deploying.

# Modify the Styling of the Themes

## Exercise Goals

- Customize and modify variables
  - Modify Clay Base variables
  - Add custom theme variables to the theme
- Modify components and global styles
  - Modify Clay Base partials
  - Add custom CSS and import it into the source file
- Add custom images

## Add the SCSS File Structure to the Theme Source

1. **Copy** the contents of the `exercise-src\css` folder.
2. **Paste** the files into your `livingstone-fjord-theme\src\css` folder.

## Modify Clay Variables to Fit Branding

1. **Drop** `_clay_variables.scss` from `livingstone-fjord-theme\src\css` into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 01-custom-clay-variables here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `01-custom-clay-variables` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```
$body-bg: white;

$bright-color: #1865FB;

$complementary-color: #869CAD;

$dark-color: #242529;

$light-color: #E7E7ED;

$solid-color: #47647A;


$portlet-decorate-bg: #FFF;

$portlet-decorate-border: 1px solid $light-color;
```

## Add Color Variables for the Theme

1. **Drop** `_colors.scss` from *livingstone-fjord-theme* | *src* | *css* | *partials* | *variables* into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 02-colors-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `02-colors-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
$fjord-primary: #5657A1;

$fjord-info: #7BC6DC;

$fjord-secondary: #BDCADB;

$fjord-success: #57D192;
```

## Import the Variable Partial Files

1. **Drop** `_variables.scss` from *livingstone-fjord-theme* | *src* | *css* | *partials* into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 03-variables-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `03-variables-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
@import "variables/colors";
```

## Customize the Portlet Variables

1. **Drop** `_variables_custom.scss` from *livingstone-fjord-theme* | *src* | *css* | *portlet* into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 04-portlet-variables-custom-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `04-portlet-variables-custom-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
$portlet-header-margin-bottom: map-get($spacers, 5);
```

```
$portlet-topper-bg: #F1F5FF;
```

```
$portlet-topper-border: #D7E5FF;
```

```
$portlet-topper-color: map-get($theme-colors, primary);
```

```
$portlet-topper-link-color: map-get($theme-colors, primary);
```

```
$portlet-topper-link-hover-color: map-get($theme-colors, primary);
```

```
$portlet-content-border-radius: 0 0 8px 8px;
```

```
$portlet-topper-border-radius: 4px 4px 0 0;
```



## Customize the Button Component

1. **Drop** the `_buttons.scss` file found in `livingstone-fjord-theme\src\css\partials` into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 05-buttons-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `05-buttons-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
.fjord-btn-primary {  
    @include button-variant($fjord-primary, $fjord-primary);  
}
```

## Add Custom Styling to the Footer

1. **Drop** the `_footer.scss` file found in `livingstone-fjord-theme\src\css\partials` into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 06-footer-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `06-footer-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
.footer {
  a {
    color: inherit;

    &:hover,
    &:focus {
      color: white;
    }
  }
}

// Insert snippet 10-social-media-footer-scss here
}
```

## Import Partial to Custom.scss

1. **Drop** the `_custom.scss` file found in `livingstone-fjord-theme\src\css` into the *Visual Studio Code* editor.
2. **Press *Enter*** below the `/* inject:imports */` note.
3. **Type `lfr`** to view the available code snippets.
4. **Choose** the `07-custom-scss-imports` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
@import "partials/variables";

@import "partials/backgrounds";

@import "partials/buttons";

@import "partials/footer";

@import "partials/header";

@import "partials/texts";

@import "portlet/portlet_decorator";
```

## Add Theme Images

1. **Copy** the contents of the `images` folder from *exercise-src*.
2. **Paste** the contents into *livingstone-fjord-theme/src/images*.
  - You will need to replace the thumbnail image.

## Deploy the Theme to See the Styling Changes

1. **Run** `npm run deploy` in the *Command Line* or the *Terminal*.
  - Make sure you are in the `liferay/livingstone-fjord-theme` directory
  - If you're already running `gulp watch`, this isn't needed.
2. **Open** your browser.
  - You should already have a tab open at `localhost:8080`
3. **Click** to *Refresh* the page after the theme changes have finished deploying.

*Note: Now you should be able to see all the styling changes we made to the platform. We changed the theme colors, sizes of various elements, as well as the image for our theme. If you have time, feel free to change some of these things and customize the theme however you'd like.*

## Add the Livingstone Logo

1. **Click** *Site Administration > Site Builder > Pages* in the *Menu* for the *Livingstone Hotels & Resorts* site.
2. **Click** the gear icon next to *Public Pages*.
3. **Click** to expand the *Logo* section near the bottom of the page.
4. **Click** *Change > Select*.
5. **Choose** the `livingstone-logo.png` file from the *exercise* folder.
6. **Click** *Done*.
7. **Click** to set the *Show Site Name* slider to *No*.
8. **Click** *Save*.

---

## Bonus Exercise

1. Change the theme colors and sizes of various elements using the files we created or updated in the exercise above. Be creative and make it look how you'd like it to look.

# Add Custom JavaScript to a Theme

## Exercise Goals

- Add JavaScript to the theme
  - Add DOM variables
  - Add the Sign-in Modal
  - Add the Click event for closing the nav

## Add Main.js to the Theme

1. **Copy** the `main.js` from the `exercise-src\js` folder.
2. **Paste** it into your `livingstone-fjord-theme/src/js` folder.

## Set DOM Variables and the Sign-In Widget

1. **Drop** the `main.js` from `livingstone-fjord-theme\src\js` into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 01-variables-and-sign-in here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `01-variables-and-sign-in` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```

var BODY = A.getBody();

var signIn = A.one('#sign-in');

if (signIn && signIn.getData('redirect') !== 'true') {

    signIn.plugin(Liferay.SignInModal);

}

var fullScreenToggleIcon = A.one('.full-screen-navigation #banner .navbar-toggler');

```

## Add the Click Event for Closing the Navigation

1. **Click** to highlight the `// Insert snippet 02-close-nav-click here comment`.
2. **Type** `lfr` to view the available code snippets.
3. **Choose** the `02-close-nav-click` snippet.
4. **Save** the file.

- Alternatively, you can type and save the following:

```

if (fullScreenToggleIcon) {

    fullScreenToggleIcon.on(

        'click',

        function(event) {

            BODY.toggleClass('main-nav-opened', event.currentTarget.hasClass('collapsed'));

        }

    );

}

```

## Deploy the Theme to See the JavaScript Changes

1. **Run** `npm run deploy` in the *Command Line* or *Terminal*.
  - If you're already running `gulp watch`, this isn't needed.
2. **Open** your browser.
  - You should already have a tab open at `localhost:8080`
3. **Click** to *Refresh* the page after the theme changes have finished deploying.



# Configure the Theme to Provide Administrative Options

## Exercise Goals

- Add Portlet Decorators to the theme
  - Add the Configuration files
  - Add styling for the Portlet Decorators
- Add Custom Theme Settings
  - Add Theme Settings
  - Create Theme Setting Variables
  - Add FreeMarker Files and Styling for Theme Settings

## Add WEB-INF Files to the Theme

1. **Copy** the remaining contents from the `exercise-src/WEB-INF` folder.
  - The `liferay-look-and-feel.xml` file was added in Exercise 1: Create the Livingstone Fjord Theme.
2. **Paste** the files into the `livingstone-fjord-theme/src/WEB-INF` folder.
  - You'll need to replace the existing `liferay-plugin-package.properties`.

## Add Portlet Decorators

1. **Drop** the `liferay-look-and-feel.xml` file from `livingstone-fjord-theme/src/WEB-INF` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 01-portlet-decorators here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `01-portlet-decorators` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:



```
<portlet-decorator id="barebone" name="Barebone">

  <portlet-decorator-css-class>portlet-barebone</portlet-decorator-css-class>

</portlet-decorator>

<portlet-decorator id="borderless" name="Borderless">

  <portlet-decorator-css-class>portlet-borderless</portlet-decorator-css-class>

</portlet-decorator>

<portlet-decorator id="decorate" name="Decorate">

  <default-portlet-decorator>true</default-portlet-decorator>

  <portlet-decorator-css-class>portlet-decorate</portlet-decorator-css-class>

</portlet-decorator>

<portlet-decorator id="solid" name="Solid">

  <portlet-decorator-css-class>portlet-solid</portlet-decorator-css-class>

</portlet-decorator>
```

## Add Sass Styling for the Portlet Decorator

1. **Drop** the `_portlet_decorator.scss` from *livingstone-fjord-theme* | *src* | *css* | *portlet* into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 09-portlet-decorator-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `09-portlet-decorator-scss` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```

.portlet-decorate {

    .portlet {

        margin-bottom: 0;

    }

    .portlet-content {

        border-width: 0;

        padding: 0;

    }

}

.portlet-solid .portlet-content {

    background: #FFF;

    border: 1px solid #DEEEEEE;

}

.portlet-barebone .portlet-content {

    padding: 0;

}

```

## Add Custom Theme Settings to the Theme XML File

1. **Drop** the `liferay-look-and-feel.xml` from `livingstone-fjord-theme\src\WEB-INF` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 02-theme-settings here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `02-theme-settings` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```
<setting configurable="true" key="facebook-icon" type="checkbox" value="true" />
<setting configurable="true" key="facebook-icon-link-url" value="http://www.facebook.com'
<setting configurable="true" key="twitter-icon" type="checkbox" value="true" />
<setting configurable="true" key="twitter-icon-link-url" value="http://www.twitter.com" ,
```

## Add Custom Variables in the Init\_Custom File

1. **Drop** the `init_custom.ftl` from `livingstone-fjord-theme\src\templates` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 06-init-custom-settings here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `06-init-custom-settings` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```
facebook_icon = getterUtil.getBoolean(themeDisplay.getThemeSetting("facebook-icon"))
facebook_icon_link_url = themeDisplay.getThemeSetting("facebook-icon-link-url")
twitter_icon = getterUtil.getBoolean(themeDisplay.getThemeSetting("twitter-icon"))
twitter_icon_link_url = themeDisplay.getThemeSetting("twitter-icon-link-url")
```

## Add Social Media Icons to the Theme Footer

1. **Drop** the `footer.ftl` from `livingstone-fjord-theme\src\templates` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 07-social-portal-normal here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `07-social-portal-normal` snippet.
5. **Save** the file.
  - Alternatively, you can type and save the following:

```
<div class="ml-md-auto mt-3 mt-md-0">
    <#include "${full_templates_path}/social_media.ftl" />
</div>
```

## Add the Code to the Social Media FreeMarker File

1. **Drop** the `social_media.ftl` from `livingstone-fjord-theme|src|templates` into the *Visual Studio Code* editor.
2. **Click** to highlight the `<!-- Insert snippet 08-social-media-footer here -->` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `08-social-media-footer` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
<#if facebook_icon || twitter_icon>

    <ul class="list-inline mb-0">

        <#if facebook_icon>

            <li class="list-inline-item">

                <a class="bg-white bg-white-10 d-block fjord-social fjord-text-primary
                    rounded-circle text-center text-decoration-none text-primary" href="{facebook_url}">
                    <span class="fa fa-facebook icon-monospaced"></span>
                </a>

            </li>

        </#if>

        <#if twitter_icon>

            <li class="list-inline-item">

                <a class="bg-white bg-white-10 d-block fjord-social fjord-text-info
                    rounded-circle text-center text-decoration-none text-info" href="{twitter_url}">
                    <span class="icon-monospaced fa fa-twitter-square"></span>
                </a>

            </li>

        </#if>

    </ul>

</#if>
```

## Add Sass Styling to Social Media Icons

1. **Drop** the `_footer.scss` from `livingstone-fjord-theme\src\css\partial` into the *Visual Studio Code* editor.
2. **Click** to highlight the `// Insert snippet 10-social-media-footer-scss here` comment.
3. **Type** `lfr` to view the available code snippets.
4. **Choose** the `10-social-media-footer-scss` snippet.
5. **Save** the file.

- Alternatively, you can type and save the following:

```
.fjord-social {  
    @include size(64px);  
    line-width: 64px;  
  
    &:hover,  
    &:focus {  
        color: white !important;  
    }  
}
```

## Deploy the Theme to See the Changes

1. **Run** `npm run deploy` in the *Command Line* or *Terminal*.
  - If you're already running `gulp watch`, this isn't needed.
  - You may need to reselect the Livingstone Fjord theme, if the default theme is still shown after refreshing the page.
2. **Open** your browser.
  - You should already have a tab open at `localhost:8080`
3. **Click** to *Refresh* the page after the theme changes have finished deploying.

# Adding Style Books

## Exercise Goals

- Create a style book in the UI

## Change the Theme to Classic

1. **Go to** `localhost:8080` in your browser.
  - Log in to the portal as an admin if you have been logged out.
2. **Open the** `Site Administration Menu`.
3. **Go to** `Site Builder > Pages`.
4. **Click** the gear icon next to *Public Pages*.
5. **Click** the *Change Current Theme* button near the bottom of the page.
6. **Choose** the *Classic* theme.
7. **Click** *Save* at the bottom of the page.

## Create a Style Book

1. **Go to** `Design > Style Books` in the `Site Administration Menu`.
2. **Click** the *Add* button to create a new Style Book.
3. **Type** `Livingstone BIG` as the *Name*.
4. **Click** the drop-down at the top where it says `Color System`.
5. **Choose** the `Typography` category.
6. **Find** the field *Heading 1 Font Size*.
7. **Choose** the *5rem* value.
8. **Click** *Publish* to complete the Style Book.
9. **Click** *OK* in the pop-up.

## Add a Content Page and Use the Style Book

1. **Go to** `Site Builder > Pages` in the *Site Administration* menu.
2. **Click** the *Add* button to create a new page.
3. **Choose** *Public Page* from the drop-down.
4. **Choose** *Blank* as the page type from the Basic Templates.
5. **Type** `Style Book` as the *Name*.
6. **Drag** and drop the *Heading* fragment onto the canvas.
7. **Click** the *Page Design Options* (paint roller) icon in the content pages toolbar.
8. **Choose** the *Style Book* tab.
9. **Select** the *Livingstone BIG* Style Book.
  - The heading should change to a much larger font.

---

## Bonus Exercises

1. Create a new style book that changes the color-system of the theme based on what you think looks best.

## Create Site Branding Module Quiz

1. Which of the following settings and packages can be configured in your theme? (Choose two.)
  - A. Theme Settings
  - B. Widget Styles
  - C. Bundled Color Schemes
  - D. Portlet Decorators
2. Which of the following are the primary source files to customize in a Theme? (Choose three.)
  - A. `main.scss`
  - B. `portal_normal.ftl`
  - C. `liferay-plugin-package.properties`
  - D. `main.js`
  - E. `_custom.scss`
3. In Liferay DXP 7.3, the Clay base is built on Bootstrap and converts LESS to SASS.
  - A. True
  - B. False
4. Which of the following is NOT a configurable setting that can be added to a theme?
  - A. Bundling Layout Templates
  - B. Portlet Decorators
  - C. Theme Settings
  - D. Themelets
  - E. Color Schemes
5. Themes can include resources like embedded widgets, web content, and documents.
  - A. True
  - B. False



## Answer Key

1. A, D
2. B, D, E
3. True
4. D
5. True