

Using Advanced Custom Fields to Create Custom Blocks

Jason Lawton

jason@jasonlawton.com

WordPress Developer

MCS

What we will cover

- Why create a block using ACF?
- What we want to accomplish
- Creating a child theme
- Block registration
- Creating the block in ACF
- Block template coding
- Using the block in the editor
- Final result

Why create a block using ACF?

- You are comfortable using ACF
- You have good knowledge of PHP, but are not strong in Javascript
- You have an idea for a layout for your site, but can't find the right block, or combination of blocks that fits your needs

What do we want to make?

Split image block

- **Image**
- **Title**
- **Subtitle**
- **Content**
- **Call to action**

What else do we want to make?

Latest posts block

- **Title**
- **Content**
- **List of posts**
 - **Number of posts**
 - **Category**

Anything else?

Slider block

- **Gallery of images**
- **Open image in a modal window**

This is a more advanced block that loads javascript libraries for showing the gallery of images in a slider.

How are we going to make them?

- Create a child theme
- Add PHP code to register the ACF block
- Set up the fields in ACF
- Code the PHP template
- Use the ACF block in the editor
- View it on the front end of the site

Creating a child theme

We're going to use the normal twenty twenty-one theme

To create a child theme, use a plugin like Child Theme Configurator or Child Theme Generator



Child Theme Configurator

[Install Now](#)

When using the Customizer is not enough - Create a child theme from your installed...

[More Details](#)

By [Lilaea Media](#)



300,000+ Active Installations

Last Updated: 3 months ago

Untested with your version of WordPress



Child Theme Generator

[Install Now](#)

Child Theme Generator is a WordPress plugin, useful to generate a child themes.

[More Details](#)

By [Serafino Corriero](#)



40,000+ Active Installations

Last Updated: 3 months ago

✓ Compatible with your version of WordPress

Block registration

Now that we have created a child theme, we can edit the functions.php file

Add the following code to the functions.php file

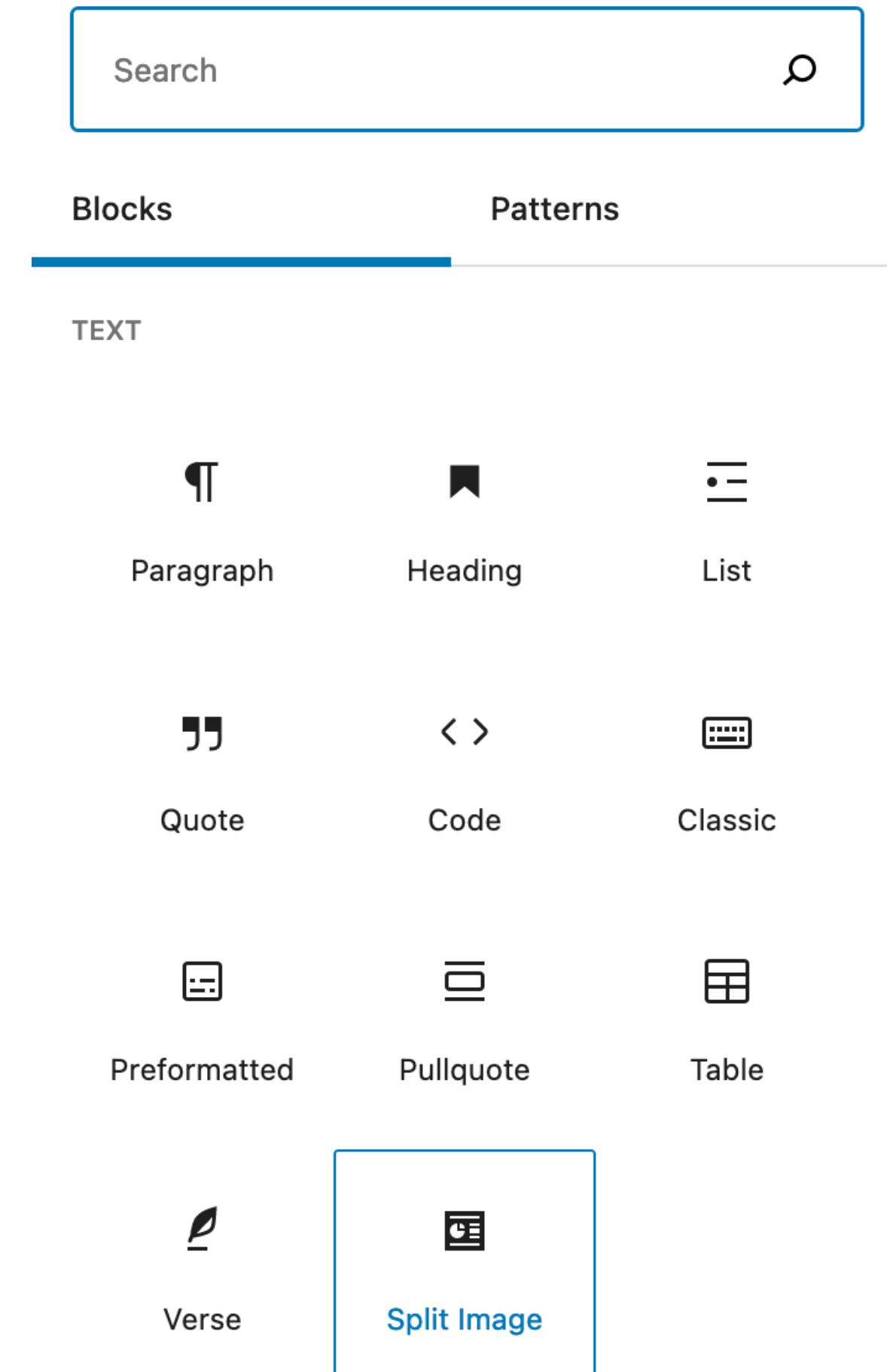
```
add_action('acf/init', 'my_acf_blocks_init');
function my_acf_blocks_init() {
    // Check if function exists.
    if( function_exists('acf_register_block_type') ) {
        // Register a split image block.
        acf_register_block_type(array(
            'name'          => 'jhl-split-image-block',
            'title'         => __('Split Image'),
            'description'   => __('Block with an image on one side, content on the other'),
            'render_template' => 'template-parts/blocks/split-image/split-image.php',
            'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/split-image/split-image.css',
            'category'       => 'layout',
            'icon'           => 'analytics'
        ));
    }
}
```

Full documentation is available at [ACF register block type](#)

Block registration

At this point we can go to the block editor and see that the block is registered correctly and shows up in the list of available blocks.

However if we click on it, nothing happens because the block is not hooked up to ACF yet.



Creating the block in ACF

Creating a block using ACF is the same as using ACF in general.

Break down the block into a form that the user will fill out.

The difference is that the user can place the block wherever they want, and reuse it multiple times on the page.

- Image
- Title
- Subtitle
- Content
- Button text
- Button link

Creating the block in ACF

Order	Label	Name	Type
1	Image	image	Image
2	Title	title	Text
3	Subtitle	subtitle	Text
4	Content	content	Wysiwyg Editor
5	Button text	button_text	Text
6	Button link	button_link	Url
			+ Add Field

Creating the block in ACF

Set the appearance to "Block" and the name of the block we specified earlier "Split Image Block".

The screenshot shows the 'Location' settings in the Advanced Custom Fields (ACF) plugin. On the left, there's a sidebar with a 'Rules' section containing instructions to create rules for edit screens. The main area has a heading 'Show this field group if' followed by a conditional logic builder. The logic is set to 'Block is equal to Split Image Block and' (with an 'or' option below it). A blue 'Add rule group' button is visible at the bottom of the logic builder.

Location

Rules
Create a set of rules to determine which edit screens will use these advanced custom fields

Show this field group if

Block is equal to Split Image Block and

or

Add rule group

Block registration

```
acf_register_block_type(array(  
    'name'          => 'jhl-split-image-block',  
    'title'         => __('Split Image'),  
    'description'   => __('Block with an image on one side, content on the other'),  
    'render_template' => 'template-parts/blocks/split-image/split-image.php',  
    'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/split-image/split-image.css',  
    'category'       => 'layout',  
    'icon'           => 'analytics'  
));
```

Block registration

```
acf_register_block_type(array(
    'name'          => 'jhl-split-image-block',
    'title'         => __('Split Image'),
    'description'   => __('Block with an image on one side, content on the other'),
    'render_template' => 'template-parts/blocks/split-image/split-image.php',
    'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/split-image/split-image.css',
    'category'       => 'layout',
    'icon'           => 'analytics',
    'mode'           => 'edit',
));

```

Block registration

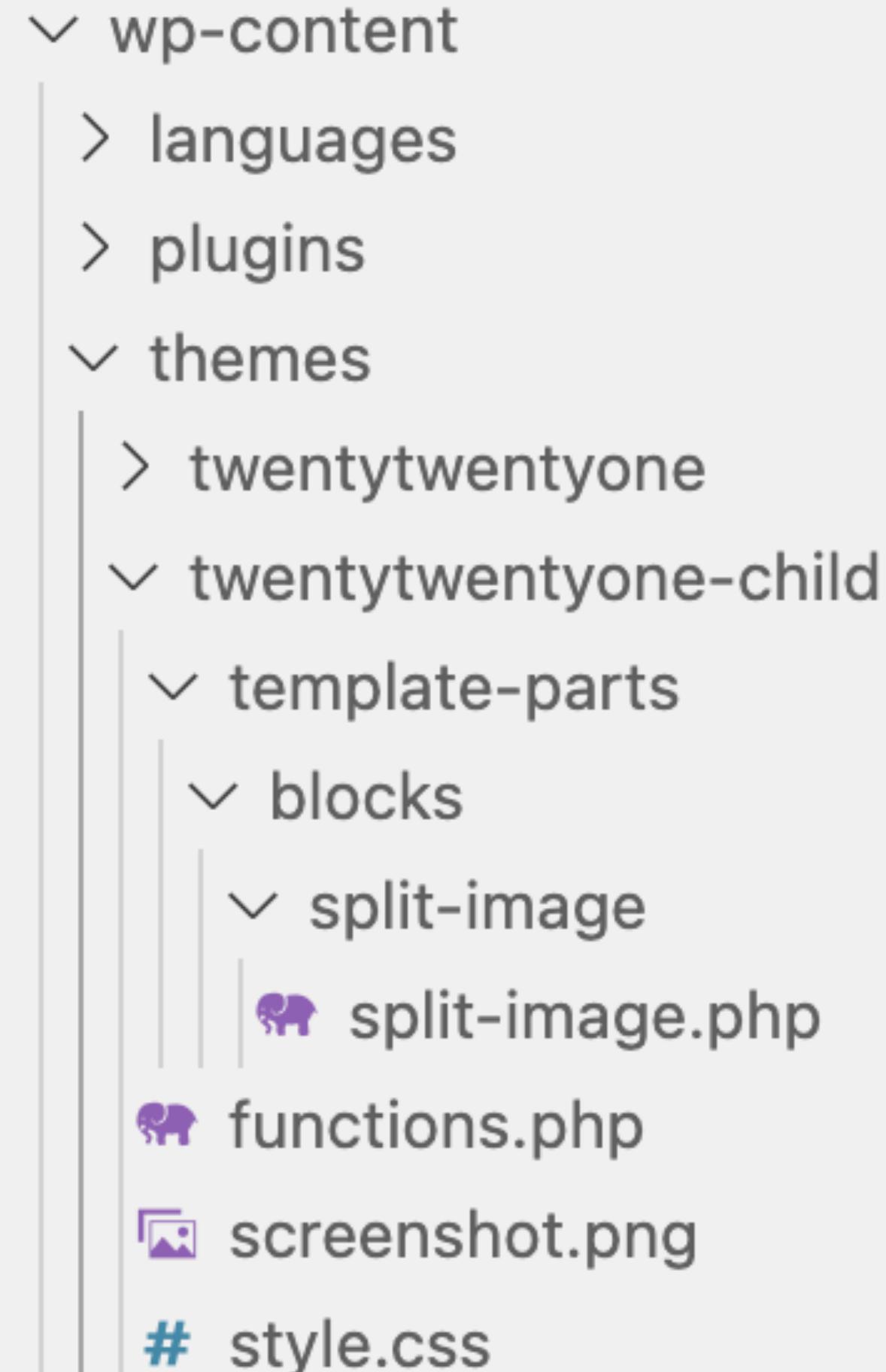
```
acf_register_block_type(array(  
    'name'          => 'jhl-split-image-block',  
    'title'         => __('Split Image'),  
    'description'   => __('Block with an image on one side, content on the other'),  
    'render_template' => 'template-parts/blocks/split-image/split-image.php',  
    'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/split-image/split-image.css',  
    'category'       => 'layout',  
    'icon'           => 'analytics',  
    'mode'           => 'edit',  
));
```

Block template coding

Create the folder structure in your child theme

template-parts/blocks/
split-image/

Create a file called split-image.php



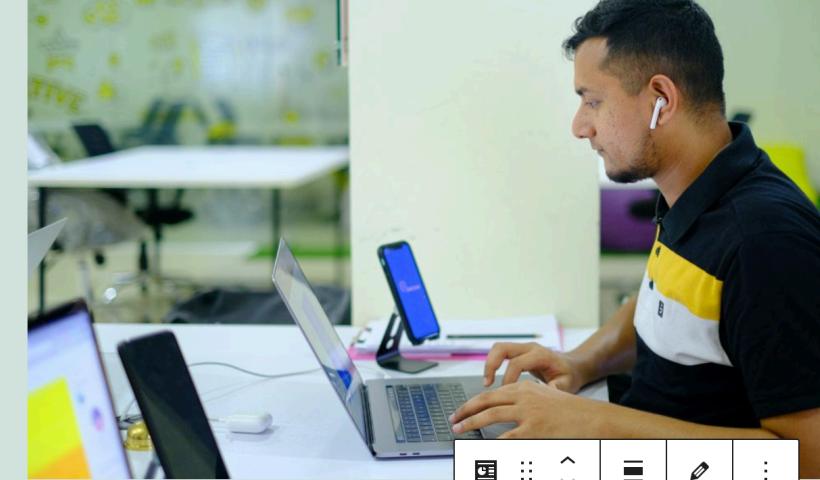
Block template coding

```
<?php
$acf_fields = get_fields();
$align_class = $block['align'] ? 'align' . $block['align'] : '';
?>
<div class="split-image-wrapper <?php echo $align_class; ?>">
  <div class="split-image-text">
    <div class="split-image-title"><?php echo $acf_fields['title']; ?></div>
    <div class="split-image-subtitle"><?php echo $acf_fields['subtitle']; ?></div>
    <div class="split-image-content"><?php echo $acf_fields['content']; ?></div>
    <div class="split-image-button">
      <a href="<?php echo $acf_fields['button-link']; ?>"><?php echo $acf_fields['button-text']; ?></a>
    </div>
  </div>
  <div class="split-image-image" style="background-image:url(<?php echo $acf_fields['image']; ?>);"></div>
</div>
```

Split image block in the editor

Switch to draft Preview Update :

+ ⌛ ⓘ ⚡



Working hard or hardly working?

Block with an image on one side, content on the other

No image selected Add Image

Title

Subtitle

Content

Add Media Visual Text

Paragraph B I

Text

19

Block with an image on one side, content on the other

No image selected Add Image

Title

Subtitle

Content

Add Media Visual Text

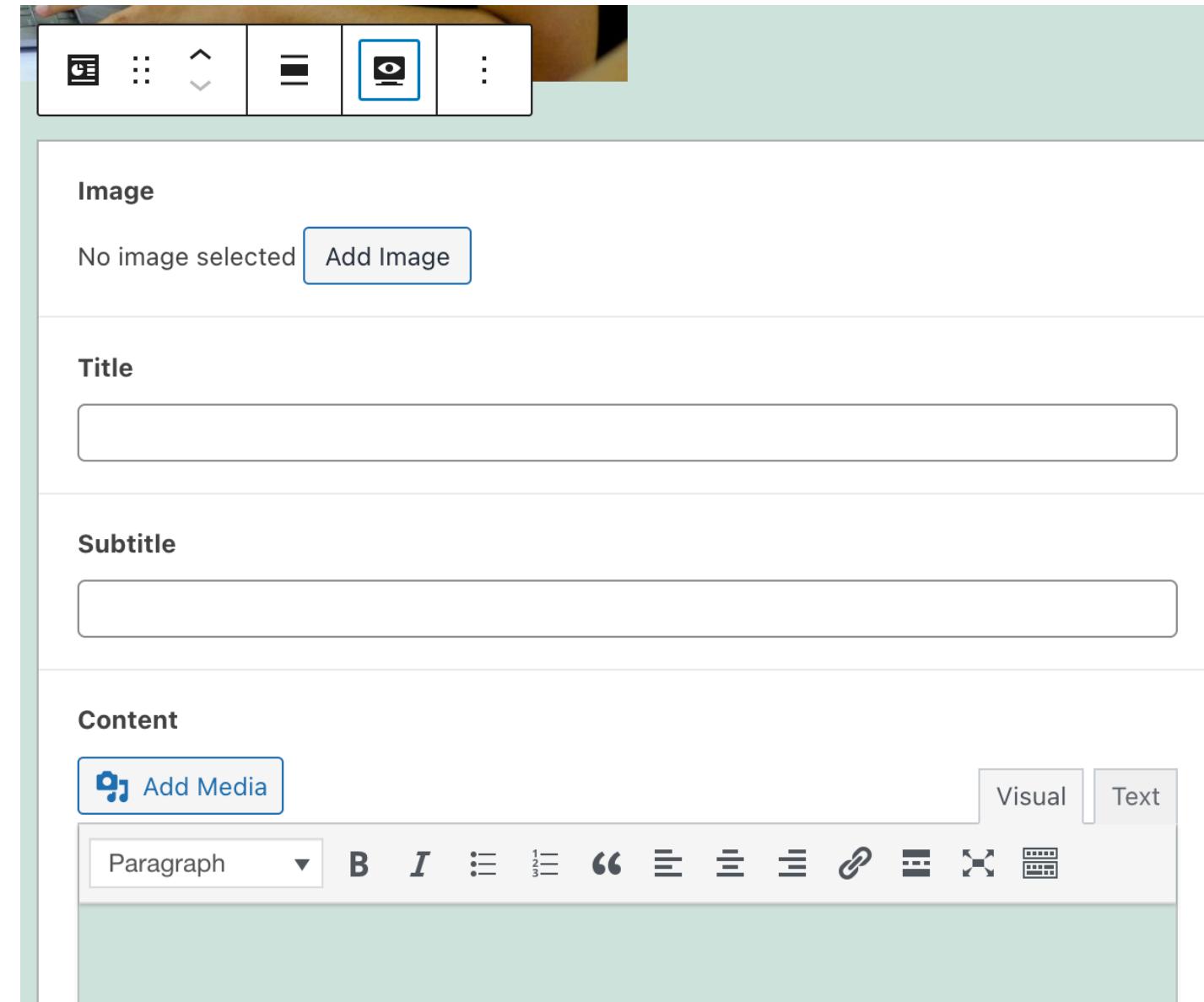
Paragraph B I

Text

Split image block in the editor



Split image block in the editor



Block registration

```
acf_register_block_type(array(  
    'name'          => 'jhl-split-image-block',  
    'title'         => __('Split Image'),  
    'description'   => __('Block with an image on one side, content on the other'),  
    'render_template' => 'template-parts/blocks/split-image/split-image.php',  
    'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/split-image/split-image.css',  
    'category'       => 'layout',  
    'icon'           => 'analytics',  
    'mode'           => 'edit',  
));
```

Split image block final result in editor

Work from anywhere!

As long as you're working

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

[Apply now!](#)



Split image block final result on site

Work from anywhere!

As long as you're working

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

[Apply now!](#)



Latest posts block

This block will support the following fields

- **Title**
- **Subtitle**
- **Image**
- **Content**
- **List of posts from the posts content type**
 - **Allow the user to specify how many posts**
 - **Allow the user to specify a category**

Block registration

```
acf_register_block_type(array(
    'name'          => 'jhl-latest-posts-block',
    'title'         => __('Latest Posts'),
    'description'   => __('Block with an title, content, and latest posts'),
    'render_template' => 'template-parts/blocks/latest-posts/latest-posts.php',
    'enqueue_style'  => get_stylesheet_directory_uri() . '/template-parts/blocks/latest-posts/latest-posts.css',
    'category'       => 'layout',
    'icon'           => 'analytics',
    'mode'           => 'edit',
));
});
```

Creating the block in ACF

Order	Label	Name	Type
1	Title Edit Duplicate Move Delete	title	Text
2	Subtitle	subtitle	Text
3	Image	image	Image
4	Content	content	Wysiwyg Editor
5	Number of Posts to show	number_of_posts_to_show	Number
6	Post Category	post_category	Taxonomy

[+ Add Field](#)

Location ▲

Rules
Create a set of rules to determine which edit screens will use these advanced custom fields

Show this field group if

Block is equal to Latest Posts and

or

Block template coding

```
$latest_posts_args = [
    'post_type' => 'post',
];
if ( $acf_fields['number_of_posts_to_show'] ) {
    $latest_posts_args['posts_per_page'] = $acf_fields['number_of_posts_to_show'];
}
if ( $acf_fields['post_category'] ) {
    $latest_posts_args['category__in'] = $acf_fields['post_category'];
}
$latest_posts = new WP_Query( $latest_posts_args );
```

Block template coding

```
<?php if ( $latest_posts->have_posts() ) { ?>
    <div class="latest-posts-list <?php echo $align_class; ?>">
        <?php while ( $latest_posts->have_posts() ) {
            $latest_posts->the_post();
        ?>
            <div class="latest-post">
                <div class="latest-post-title">
                    <a href="php the_permalink(); ?&gt;"&gt;&lt;?php the_title(); ?&gt;&lt;/a&gt;
                &lt;/div&gt;
                &lt;div class="latest-post-content"&gt;&lt;?php the_excerpt(); ?&gt;&lt;/div&gt;
            &lt;/div&gt;
        &lt;?php } ?&gt;
    &lt;/div&gt;
&lt;?php } ?&gt;</pre
```

Latest posts block in the editor

Number of Posts to show

 ^ ▼

Post Category

Food ▼

Select a category to only bring up posts in that category, leave blank to use "All" categories.

Latest posts block on site

Food Posts

yummy yummy

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.



Animi harum culpa laboriosam ut odit

Quo ut sit quasi ducimus. neque dolorem necessitatibus labore sit a rerum. Quo quas et itaque architecto Nam corrupti dolor quia et similique. Ut ipsum ipsam totam eveniet. Doloribus et nulla aliquam. Quia numquam eos assumenda.

Aut velit repudiandae ratione

Autem est dolor est id qui. Voluptates molestiae facilis et ut eveniet recusandae Ex magnam id ipsa qui officia quia. Vel in voluptas Voluptatibus corporis voluptatem est voluptas fugit. voluptatem repellendus qui est aliquid nam.

Voluptatem tempore voluptatem repellat eveniet et ratione

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Slider block

Display a gallery of images in a slider component.

While the block fields are simple - a gallery field and a checkbox to open the images in a lightbox, the block code is more complex because it loads scripts and styles to facilitate the slider.

Block registration

```
acf_register_block_type(array(
    'name'          => 'slider',
    'title'         => __('Slider'),
    'description'   => __('ACF Slider w/large image support.'),
    'render_template' => 'template-parts/blocks/slider/slider.php',
    'category'      => 'embed',
    'icon'          => 'format-gallery',
    'keywords'       => array( 'slider', 'gallery', 'carousel' ),
    'align_content'  => 'center',
    'mode'          => 'edit',
    'enqueue_assets' => function(){
        // wp_enqueue_style( 'slider-slick',
        // wp_enqueue_style( 'slider-slick',
        // wp_enqueue_style( 'slider-slick-theme',
        // wp_enqueue_script( 'slider-slick',
        // wp_enqueue_style( 'slider-slick-lightbox-theme',
        // wp_enqueue_script( 'slider-slick-lightbox',
            get_template_directory_uri() . '/template-parts/blocks/slider/slick.min.css' );
            get_template_directory_uri() . '/template-parts/blocks/slider/slick.css' );
            get_template_directory_uri() . '/template-parts/blocks/slider/slick-theme.min.css' );
            get_template_directory_uri() . '/template-parts/blocks/slider/slick.min.js', array('jquery'), '', true );
            get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.css' );
            get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.js', array('jquery','slider-slick'), '', true );
    },
));
});
```

Block registration

```
acf_register_block_type(array(
    'name'          => 'slider',
    'title'         => __('Slider'),
    'description'   => __('ACF Slider w/large image support.'),
    'render_template' => 'template-parts/blocks/slider/slider.php',
    'category'      => 'embed',
    'icon'          => 'format-gallery',
    'keywords'       => array( 'slider', 'gallery', 'carousel' ),
    'align_content'  => 'center',
    'mode'          => 'edit',
    'enqueue_assets' => function(){
        // wp_enqueue_style( 'slider-slick',
        // wp_enqueue_style( 'slider-slick',
        // wp_enqueue_style( 'slider-slick-theme',
        // wp_enqueue_script( 'slider-slick',
        // wp_enqueue_style( 'slider-slick-lightbox-theme',
        // wp_enqueue_script( 'slider-slick-lightbox',
            get_template_directory_uri() . '/template-parts/blocks/slider/slick.min.css' );
        get_template_directory_uri() . '/template-parts/blocks/slider/slick.css' );
        get_template_directory_uri() . '/template-parts/blocks/slider/slick-theme.min.css' );
        get_template_directory_uri() . '/template-parts/blocks/slider/slick.min.js', array('jquery'), '', true );
        get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.css' );
        get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.js', array('jquery','slider-slick'), '', true );
    },
));
});
```

Block registration

```
acf_register_block_type(array(
    'name'          => 'slider',
    'title'         => __('Slider'),
    'description'   => __('ACF Slider w/large image support.'),
    'render_template' => 'template-parts/blocks/slider/slider.php',
    'category'      => 'embed',
    'icon'          => 'format-gallery',
    'keywords'      => array( 'slider', 'gallery', 'carousel' ),
    'align_content' => 'center',
    'mode'          => 'edit',
    'enqueue_assets' => function(){
        wp_enqueue_style( 'slider-slick',           get_template_directory_uri() . '/template-parts/blocks/slider/slick.css' );
        wp_enqueue_style( 'slider-slick-theme',       get_template_directory_uri() . '/template-parts/blocks/slider/slick-theme.min.css' );
        wp_enqueue_script( 'slider-slick',            get_template_directory_uri() . '/template-parts/blocks/slider/slick.min.js', array('jquery'), '', true );
        wp_enqueue_style( 'slider-slick-lightbox-theme', get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.css' );
        wp_enqueue_script( 'slider-slick-lightbox',     get_template_directory_uri() . '/template-parts/blocks/slider/slick-lightbox.min.js', array('jquery','slider-slick'), '', true );
    },
));
});
```

Creating the block in ACF

Order	Label	Name	Type
1	Slider gallery	slider_gallery	Gallery
2	Open slides in lightbox	use_lightbox	True / False

+ Add Field

Location

Rules
Create a set of rules to determine which edit screens will use these advanced custom fields

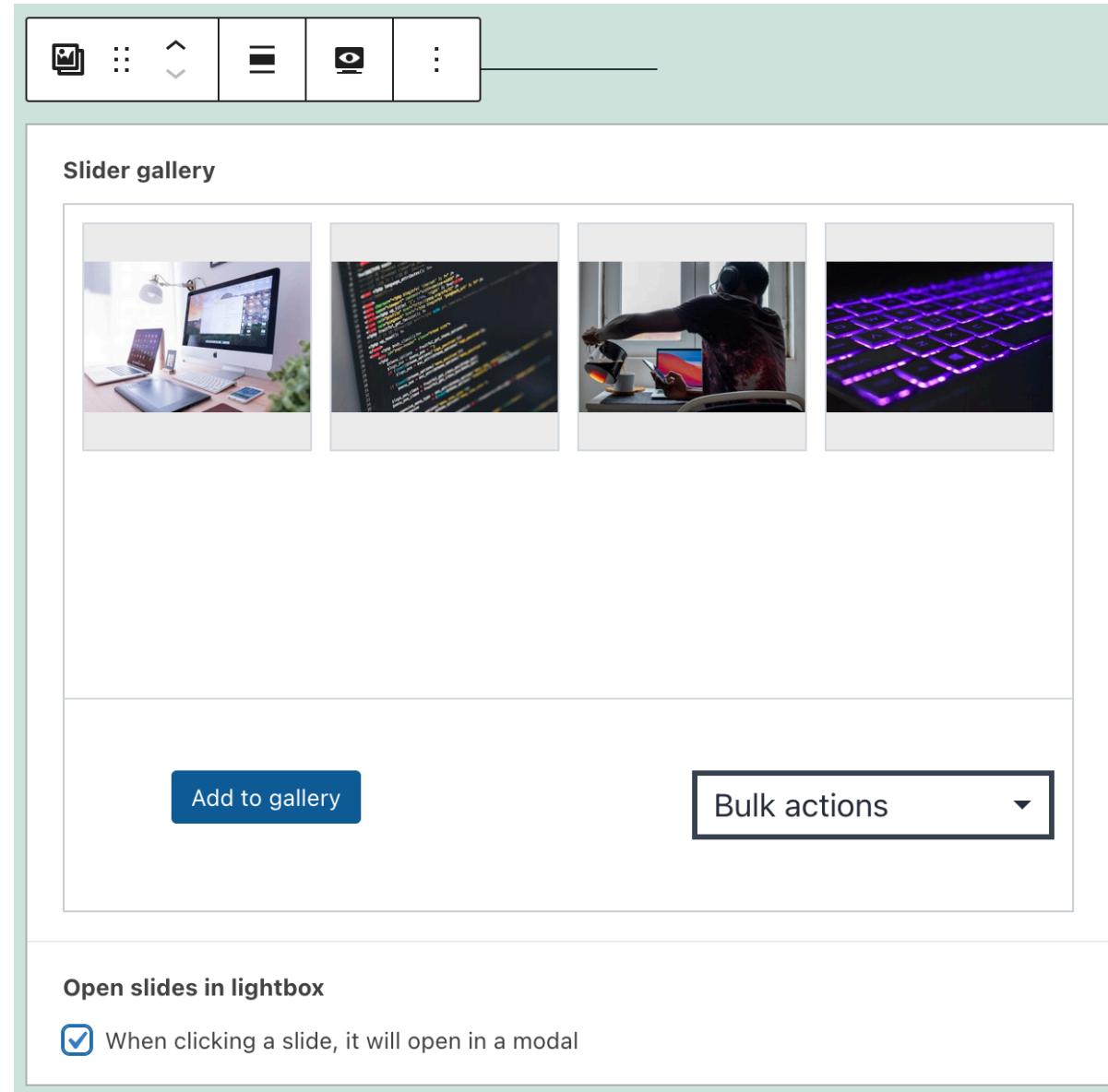
Show this field group if

Block ▾ is equal to ▾ Slider ▾ and

or

Add rule group

Slider in the block editor



Slider in on the front end



References

- <https://www.advancedcustomfields.com/resources/blocks/>
- <https://torquemag.io/2020/01/create-gutenberg-block-with-acf/>
- <https://whiteleydesigns.com/acf-gutenberg-block-examples/>

Repositories

**Presentation - [https://github.com/phoopee3/
creating-blocks-using-acf](https://github.com/phoopee3/creating-blocks-using-acf)**

**Code - [https://github.com/phoopee3/creating-
blocks-using-acf-code](https://github.com/phoopee3/creating-blocks-using-acf-code)**

Q&A

Thank you