acf_get_field_groups()

- Description
- Parameters
- Examples
  - Basic usage
  - Advanced usage
- Notes

Description

Retrieves all ACF field groups filtered by `WP_Query` args, returning them as an array.

This function should not be confused with get_fields(), which returns an array of field values for a specific post, get_field_objects(), which returns all fields and settings but doesn't separate them by field group, or acf_get_fields(), which retrieves all the fields from a specific field group.

Parameters

```
acf_get_field_groups( $filter );
```

- `$filter` *(array) (Optional)* An array of args to filter the post query by.

Examples

Basic usage

This example demonstrates how to return all field groups associated with a post ID as an array. It retrieves all the field groups, runs a filter on them, and then checks the Location Rules for each field group.

The post ID must be passed as an argument, or no field groups will be returned. You will have to replace `$post_id` with the actual ID of the post.

```
$groups = acf_get_field_groups( array( 'post_id' => $post_id ) );
```

Advanced usage

This example demonstrates how to dynamically retrieve field groups based on the current page's post type and template, and then use the retrieved field groups to generate a custom page layout based on the field groups assigned to a post.

Let's say you have a custom post type called "portfolio" and you want to display a different layout for each portfolio item based on the field groups assigned to it. You can use `acf_get_field_groups()` to retrieve the field groups and then loop through them to generate the custom layout.

```
// Get the current post ID
$post_id = get_the_ID();

// Retrieve all field groups associated with the current post
$field_groups = acf_get_field_groups(array('post_id' => $post_id));

// Loop through each field group
foreach ($field_groups as $field_group) {
    // Get the field group ID
    $field_group_id = $field_group['ID'];

    // Get the fields within the field group
    $fields = acf_get_fields($field_group_id);

    // Check if the field group has a specific layout
    if ($field_group['layout'] == 'gallery') {
        // Loop through each field and display the gallery layout
        foreach ($fields as $field) {
            if ($field['type'] == 'gallery') {
                // Display the gallery field
                the_field($field['name']);
            }
```

```
        }
    } elseif ($field_group['layout'] == 'video') {
        // Loop through each field and display the video layout
        foreach ($fields as $field) {
            if ($field['type'] == 'video') {
                // Display the video field
                the_field($field['name']);
            }
        }
    }
}
```

In this example, we first retrieve the current post ID using `get_the_ID()`. Then, we use `acf_get_field_groups()` to retrieve all field groups associated with the current post. We loop through each field group and retrieve the fields within it using `acf_get_fields()`. Finally, we check the layout of each field group and display the corresponding layout based on the field types within the group.

Notes

Exercise caution when using this function, as it is the same function ACF uses to retrieve field groups and fields.

The function caches results so the fields do not need to be retrieved from the database every time, but since the values are cached, changes to them are not updated. You will need to manually clear the cached versions if you want to change the settings of a field group or field after you've called this function.

□□□ #1
□ □□□ □□□ 2026-08-29 15:44:03 UTC
□ □□□ □□□ 2026-08-29 15:44:03 UTC