Creating Your First Custom Field

- Prerequisites
- 1. Access the Admin Panel
- 2. Basic Configuration
  - Field Group Title
  - Location Rules
  - Active
- 3. Adding Fields
  - Minimum Required Settings
    - Example: Movie Fields
- General Settings
- Validation Settings
- Presentation Settings
- Conditional Logic
  - Final Step
- For Developers

A beginner-friendly guide to adding custom fields using Secure Custom Fields (SCF).

What is a custom field?

Custom fields let you add structured data to your content. With SCF, you can attach fields like text inputs, selects, image uploads, and more to your posts, pages, or custom post types.

Prerequisites

- SCF installed and activated
- A post type (default or custom) where you'll attach your fields

- Administrator access to WordPress

You can learn more about WordPress basics here:

- Theme Basics
- Plugin Basics

1. Access the Admin Panel

- Go to **SCF → Field Groups** in your WordPress admin menu.
- Click **"Add New"** to create a new group of fields.

2. Basic Configuration

Field Group Title

Give your field group a descriptive name, like `Movie Details` or `Product Specs`.

Location Rules

Choose where this group should appear. For example:

- Post type is equal to `Movie`
- Page template is `Product Page`

This ensures your fields appear only where you need them.

Note: This fine-grained control allows you to define SCF fields once and reuse them across different post types, templates, or components.

Active

Make sure the field group is set to **Active** so it appears in the editor.

Note: Some features require the field group to be **enabled** and **exposed via the REST API**. To future-proof your configuration, we recommend setting **Show in REST** to `true` when creating your field groups.

3. Adding Fields

To start building your custom field group, click the **"Add Field"** button.

Each field requires at least a few basic settings to work correctly. All other options are optional and can be used to improve the user experience in the editor.

Minimum Required Settings

In the **General** tab, you must define the following:

- **Field Type**
  Specifies what kind of data the field will store (e.g., Text, Image, Number, Select).
- **Field Label**
  The human-readable name shown in the editor (e.g., “Movie Title”).
- **Field Name**
  A unique identifier used in the code (lowercase, no spaces; underscores or dashes allowed).

□□ Example: Movie Fields

Let’s say you want to create a group of fields for movie entries. Here's how you could configure it:

Field Label	Field Name	Field Type	Description
Movie Title	`movie_title`	Text	Stores the name of the movie
Director	`director`	Text	Stores the name of the director
Release Year	`release_year`	Number	Stores the year the movie was released
Poster Image	`poster`	Image	Upload an image file for the movie poster

> □□ These fields would be added one by one using the **Add Field** button, and each configured in the field editor panel.

General Settings

Defines the field behavior and how it is stored:

- **Field Type**
  What kind of input this field accepts (text, image, number, etc.).
- **Field Label**
  The label shown to the user in the WordPress editor.
- **Field Name**
  The code-safe name used to retrieve the value in your templates or plugins.
- **Default Value**
  A pre-filled value shown if no value is entered.

Validation Settings

Used to restrict and control the kind of data that can be entered:

- **Required**
  Makes the field mandatory.
- **Minimum / Maximum Values**
  For numeric or character-based fields. Useful for fields like `release_year`.
- **Allowed Characters / Pattern**
  Use a regular expression to restrict input format (e.g., only digits).
- **Custom Validation Message**
  Message shown if validation fails (e.g., “Please enter a valid year”).

Presentation Settings

Controls the visual appearance of the field in the editor:

- **Placeholder Text**
  Example text shown inside the field input.
- **Instructions**
  Helper text shown below the field to guide users.
- **Wrapper Attributes**
  Custom HTML attributes like class or ID for styling or JavaScript.
- **Hide Label**
  Option to hide the field label (not recommended unless styled separately).

Conditional Logic

Used to show or hide fields based on the value of other fields:

- **Enable Conditions**
  Activate conditional display rules for this field.
- **Condition Rules**
  Example: only show the “Director” field if “Content Type” equals “Movie”.

> ⚠ **Note:** Some settings may not be available for all field types. The options shown will adapt depending on the field you're configuring.

Final Step

Repeat this process to add as many fields as needed to your group. Once ready, you can assign this field group to a post type and start entering content!

For Developers

While Secure Custom Fields makes it easier to manage and display custom fields with a user-friendly interface, WordPress also includes native support for custom fields that can be managed programmatically using functions like `get_post_meta()` and `add_post_meta()`.

You can learn more about WordPress native custom fields here:

□□ WordPress Native Custom Fields

To access custom field values in your theme or plugin, use SCF functions you can learn more about SCF’s developer API in their official documentation.

□□□ #2
□ □□□ □□□ 2026-08-29 14:50:46 UTC
□ □□□ □□□ 2026-08-29 14:50:46 UTC