

カスタムフィールドで作る

カンタン投稿システム

クライアントワークに活躍する投稿システムを作ってみよう！

自己紹介

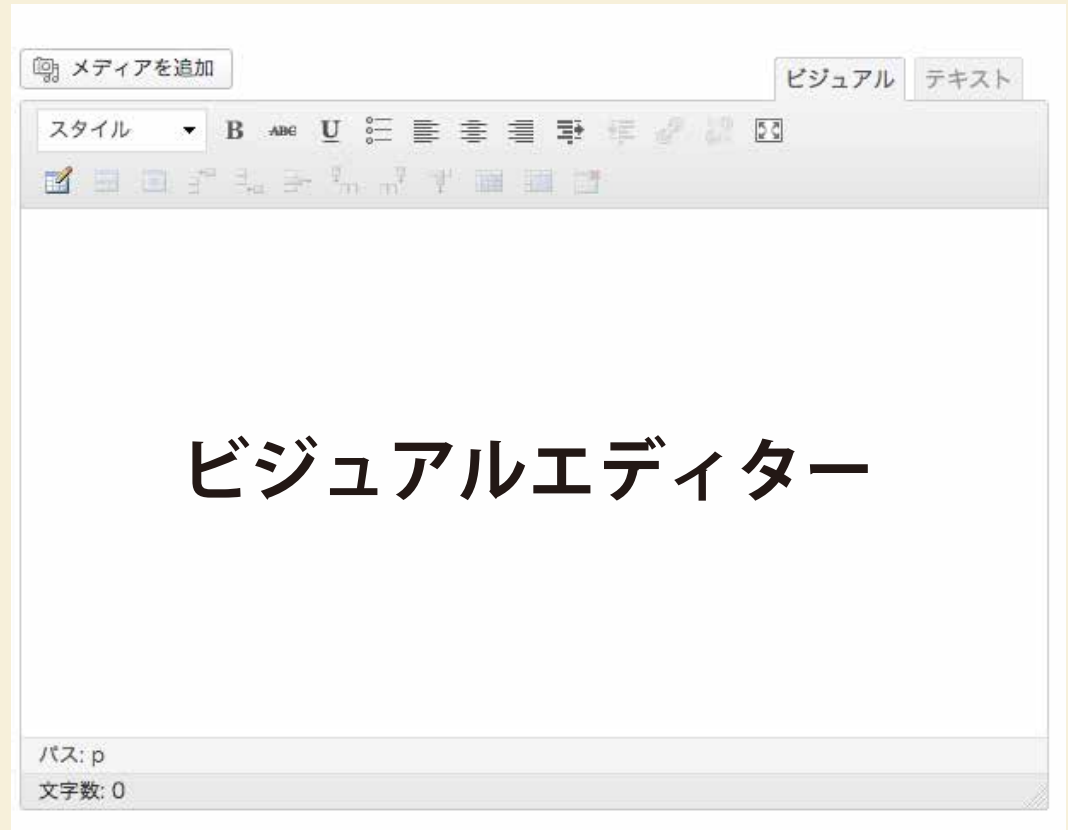


山田さおり

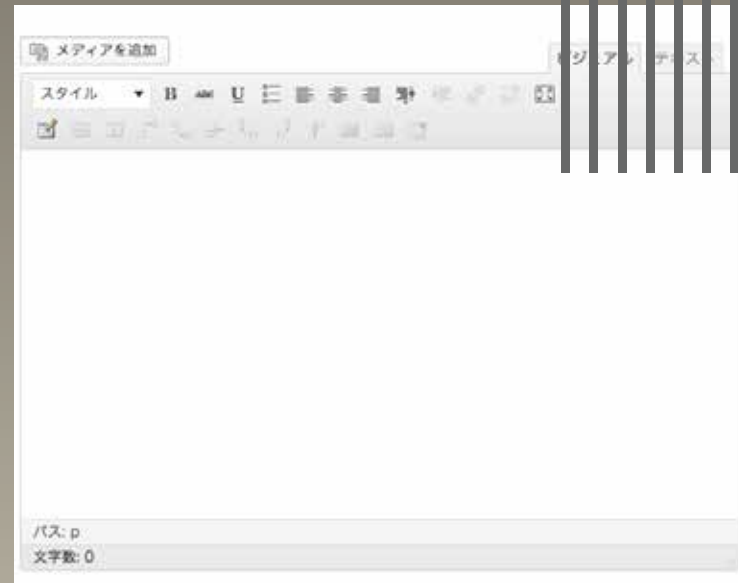
- デザイン会社で web デザイナーやっています。
- メイン業務はデザイン。
その他コーディングや CMS 導入なども手広くやっています。
- WordPress 歴は 1 年半くらい。
PHP はあまりわかりません。。。。
- facebook : saori.yamada.758
twitter : da__yama

WordPressでの記事の投稿方法

html や CSS のわからない
お客さんには、主にビジュ
アルエディターを使用して、
記事を作成してもらおうと思
いますが…



ビジュアルエディターで
凝ったレイアウトを作るのは
かなり高度なので、お客さんには難しい…



Jimdo

a-blog cms



WordPressにも Jimdo や
a-blog cms のようなウィジェットを選んで
記事を作れる機能があったらいいのにな...

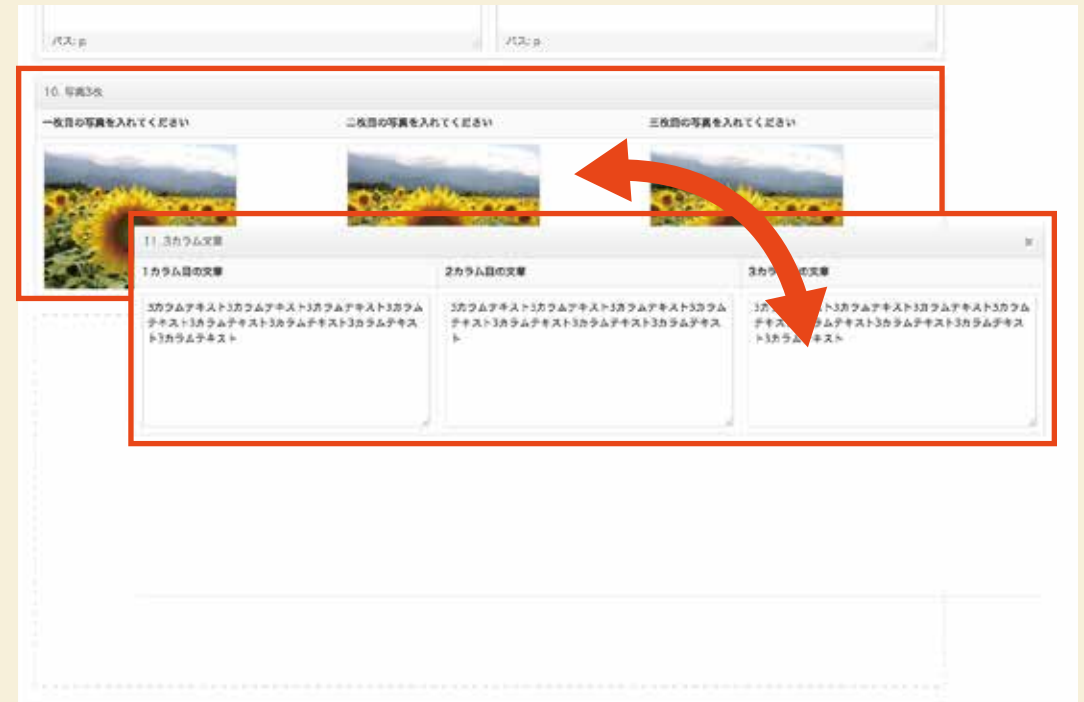
でも!

プラグインを使えば似た投稿システムが、
WordPressでも実現できます!

管理画面での操作



最下部にある青いボタンをクリックすると、レイアウトパーツが表示されます。好きなレイアウトを選んで追加します。



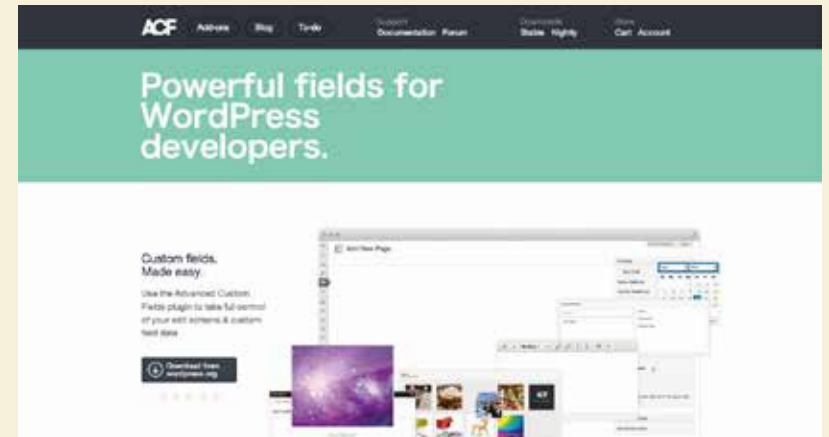
パーツごとに、ドラッグ操作で入れ替え可能です。

実現するためのプラグイン

- **Advanced Custom Fields (無料)**
- **Advanced Custom Fields の拡張プラグイン Repeater Field (有料)**
- **Advanced Custom Fields の拡張プラグイン Flexible Content Field (有料)**
- **TinyMCE Buttons Arrangement (無料、任意)**

プラグインの購入方法 & ダウンロード

1 Advanced Custom Fields プラグインの公式サイトへ行き、Repeater Field と Flexible Content Field を購入します。
(購入時にユーザー登録します)



<http://www.advancedcustomfields.com/>

2 購入ができれば、登録したアカウントでログインし、マイページの購入履歴ページから、有料プラグインのファイルがダウンロードできます。

3 ダウンロードすると、以下の zip ファイルができます。

- acf-flexible-content.zip
- acf-repeater.zip

プラグインのインストール

1 zip ファイルを解凍すると、以下のファイルができます。

- acf-flexible-content
- acf-repeater

2 上記のフォルダを、「〇〇〇〇/wp-content/plugins/」のディレクトリにアップロードしてください。

3 アップロードすると管理画面上で以下のように表示されていると思いますので、プラグインを有効にします。

<input type="checkbox"/> プラグイン	説明
<input type="checkbox"/> Advanced Custom Fields 停止 編集	Fully customise WordPress edit screen any web developer working with Word object, date picker, color picker, repea バージョン 4.1.6 作成者: Elliot Condon
<input type="checkbox"/> Advanced Custom Fields: Flexible Content Field 有効化 編集 削除	Adds the flexible content field バージョン 1.0.2 作成者: Elliot Condon
<input type="checkbox"/> Advanced Custom Fields: Repeater Field 有効化 編集 削除	Adds the repeater field バージョン 1.0.1 作成者: Elliot Condon
<input type="checkbox"/> Akismet 有効化 編集 削除	Used by millions, Akismet is quite poss keeps your site protected from spam (Sign up for an Akismet API key, and 3)

サンプルデータについて

- **index.php**
- **customlayout.php** (今回のサンプルレイアウトを記述したもの)
- **functions.php** (画像のサイズ定義など)
- **style.css** (パーツごとのレイアウト設定)
- **editor-style.css** (リセット用やスタート用の CSS の記述)
- **customlayout.xml** (今回ご紹介したレイアウトのカスタムフィールドの設定)

記述の仕方

```
<?php while(the_flexible_field(" フィールド名 ")): ?>
```

```
<?php if(get_row_layout() == " パーツごとのフィールド名 "): ?>
```

```
<?php the_sub_field(" パーツごとのサブフィールド名 "); ?>
```

1 パーツ

```
<?php elseif(get_row_layout() == " パーツごとのフィールド名 "): ?>
```

```
<?php the_sub_field(" パーツごとのサブフィールド名 "); ?>
```

```
<?php the_sub_field(" パーツごとのサブフィールド名 "); ?>
```

1 パーツ

```
<?php elseif(get_row_layout() == " パーツごとのフィールド名 "): ?>
```

```
<?php the_sub_field(" パーツごとのサブフィールド名 "); ?>
```

1 パーツ

⋮

```
<?php endif; ?>
```

```
<?php endwhile; ?>
```

ご注意

- ⚠️ 今回使用したプラグインのバージョンは、最新のものであります。以前古いバージョンでこちらのシステムを導入したら、うまく動作しなかったため、最新バージョンのものでお試しください。
- ⚠️ レイアウトボタンを追加しすぎると、積み上げた高さで画面上からボタンがはみ出てしまいます…。レイアウトのパーツは出来るだけ少なくする事をオススメします。