

オリジナルテーマ作成について

～簡単な企業サイトを作ってみよう編～

2020.06.22

企業サイト用のオリジナルテーマ作成

まずは、ファイル階層の位置関係を把握しておきましょう。

FTPのwordpressをインストールした階層より



/wp-content/themes/



っとした説明です。

- ・使いまわしの効率を上げるのには、出来るだけ「themes」にまとめておきましょう。
- ・逆に、お客さんに編集をさせたくない場合は、これ以外の場所にファイルを作成してみましょう。
- ・ファイル名の重複がおこりやすいので、管理とバックアップは要注意！

検索ワード

wordpress オリジナルテーマ作成

PHPの基本的な作法として

「<?php ~ ?>」 がphpの基本構文です。
プログラムの終了には必ずセミコロン「;」になります。
拡張子は、「.php」になります。



っとした説明です。

- ・wordpressはphpを基本として作られています。
- ・動的プログラム、サーバサイドスクリプトを「php」を使って利用しています。
- ・サーバサイドスクリプトなので「Javascript」と違いブラウザでコードを確認出来ません。

必要最低限の構造として

index.php

全てのテンプレート階層の元になる

style.css

テーマの情報を入力する場所



っとした説明です。

◇ screenshot.png(gif jpg)

・管理画面のテーマの画像になる

・最低限の構造として「index.php」と「style.css」があればwordpressは成立します。

・立ち上げ準備中やサイト移転の状態であればこれでもよいかもしれません。

・「index.php」の注意点はこのファイル名は重複する可能性が高いので管理に気を付けましょう。

検索ワード

wordpress オリジナルテーマ作成

Style.cssに記述する

Style.css

WordPressは「style.css」のコメントアウト部分を見てテーマの情報を読み取るため、オリジナルテーマの情報はここに記述します。

必須項目は「Theme Name」だけです。

```
/*  
Theme Name: sample  
Theme URI: /www/sample_test/wp-content/themes  
Description: 専用テーマ名  
Version: 1.0  
Author: xxxxx  
Author URI: http://samples.sakura.ne.jp/sample_test/  
*/
```

検索ワード

wordpress style.css テーマ情報

インクルードを活用する方法

インクルードのタグ及びファイル

- header.php
<?php get_header(); ?>
- footer.php
<?php get_footer(); ?>
- sidebar.php
<?php get_sidebar(); ?>

自作テンプレート(例: sample.php)

```
<?php get_template_part('sample'); ?>
```

Sampleの部分をファイル名に変更します。



っとした説明です。

- 共通部分のファイルは決められているファイル名になります。
- それ以外のファイル名を使用したい場合は自作テンプレートを利用してみてください。
- タグだけあって、使わない場合は「デフォルト」の内容が表示されます。

テンプレート階層について

例えば

home.php
・トップページ

single.php
・投稿ページ

category.php
・カテゴリー別のページ一覧

404.php
・エラーページ

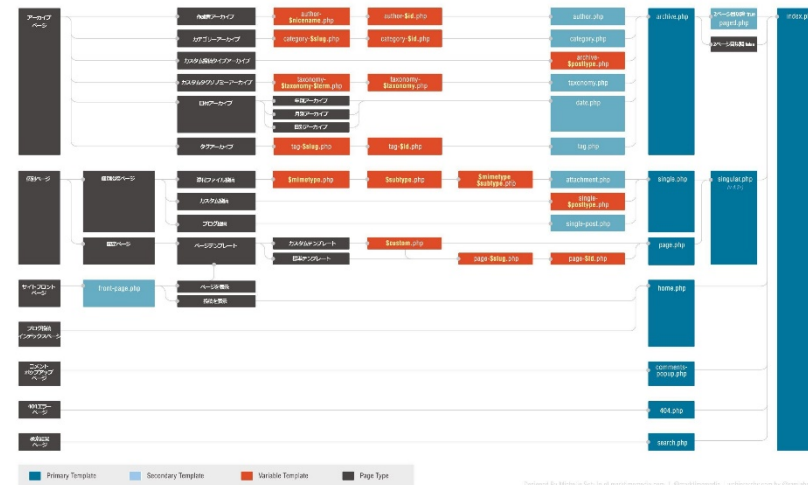
page.php
・固定ページ

tag.php
・タグ別のページ一覧

search.php
・検索結果ページ

archive.php
・アーカイブ毎のページ

date.php
・日付別のページ一覧



っとした説明です。

- ・全ての元は「index.php」になります。
- ・そのファイルがない場合は「index.php」になります。
- ・必要なファイルだけをうまく使ってコンパクトにまとめるのが運用のポイントかと思います。

検索ワード

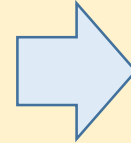
wordpress テンプレート階層

ワードプレスの独自タグ

WordPressテンプレートタグは、数多くあり全て覚えるのは困難。まずはよく利用するテンプレートタグを覚えれば問題なく作業出来るはずです。

例として:

```
<meta charset="<?php bloginfo( 'charset' ); ?>">
```



ブラウザで確認すると

```
<meta charset="UTF-8">
```



っとした説明です。

- ・基本的に使わなくても大丈夫ですがうまく使えると便利です。
- ・テンプレートの使いまわし、サーバー移転には便利です。
- ・条件分岐を使えばより便利です。

検索ワード

ワードプレスの独自タグ

よく利用するテンプレートタグ (1)

文字コード

```
<meta charset="<?php bloginfo( 'charset' ); ?>">
```

言語設定

```
<html <?php language_attributes(); ?>>
```

タイトルの設定

```
<title><?php wp_title( ' | ', true, 'right' ); ?><?php bloginfo( 'name' ); ?></title>
```

スタイルシートを読み込む

```
<link href="<?php echo get_stylesheet_uri(); ?>" rel="stylesheet">
```

テーマファイルまでのパスを読み込む

```
<?php echo get_template_directory_uri(); ?>
```

サイトタイトルを読み込む

```
<?php wp_title(); ?>
```

管理画面のキャッチフレーズ

```
<?php bloginfo( 'description' ); ?>
```

記事のパーマリンク

```
<?php the_permalink(); ?>
```

よく利用するテンプレートタグ (2)

説明文＋カスタムフィールド
※管理画面のキャッチフレーズ

```
<?php if ( $post->my_description ): ?>  
<meta name="description" content="<?php echo esc_attr( $post->my_description ); ?>">  
<?php else: ?>  
<meta name="description" content="<?php bloginfo('description'); ?>">  
<?php endif; ?>
```

キーワード＋カスタムフィールド

```
<?php if ( $post->my_keywords ): ?>  
<meta name="keywords" content="<?php echo esc_attr( $post->my_keywords ); ?>">  
<?php else: ?>  
<meta name="keywords" content="Sample,サンプル,ダウンロード">  
<?php endif; ?>
```

よく利用するテンプレートタグ (3)

<<<これ抜けているとプラグイン、悪ければテーマも動きません。>>>

※必ず入れましょう！

head内の情報

<?php wp_head(); ?>

管理ツールバー

<?php wp_footer(); ?>

よく利用するテンプレートタグ (4)

記事のパーマリンク	<code><?php the_permalink();?></code>
記事タイトル	<code><?php the_title(); ?></code>
記事内容	<code><?php the_content(); ?></code>
カテゴリ	<code><?php the_category(','); ?></code>
投稿時間	<code><?php echo get_the_date(); ?></code> <code><?php the_time(); ?></code>
アイキャッチ画像	<code><?php the_post_thumbnail(); ?></code>

TOPページの活用法

index.php

index.phpはTOPページに使用するテンプレートで、絶対に必要なファイルになります。他にもカテゴリーやタグの一覧ページなどにも使用されます。

home.php

home.phpはindex.phpよりも優先されて表示することができるTOPページのテンプレートです。両方のファイルが存在する場合、home.phpが優先されてTOPページに使用されます。

front-page.php

front-page.phpはindex.phpとhome.phpよりも優先されて表示することができるTOPページのテンプレートです。front-page.phpが一番優先度が高いTOPページのテンプレートになります。front-page.phpが存在する場合はhome.phpと違って、管理画面の表示設定で固定ページがフロントページに設定されていても影響を受けずにfront-page.phpが表示されます。

投稿ページの活用法

投稿ページの機能は、企業サイトの「新着情報」や「更新が必要」な記事、画像等があれば投稿の機能を利用すると便利です。

single.php

投稿ページのファイル



っとした説明です。

- ・記事には日付やコメント、前後の記事リンクなどが表示されます。
- ・テーマによっては関連記事が表示されたり、逆に日付などを非表示にする選択機能が備わっていたりもします。
- ・作成したページは更新した順から時系列に並んでいきます。
- ・投稿ページを利用して作成したページは必ず「投稿記事一覧」のようなアーカイブページにも表示されます。
- ・投稿ページには「カテゴリー」が設定でき、記事を分類することができます。
- ・投稿ページには「パーマリンク設定」が適用されます。

検索ワード

wordpress 投稿ページ

固定ページの活用法

固定ページは一般的に「会社案内」「お問い合わせ」「サイトマップ」など、時系列に並ぶ必要のない単体のページを作るときに使用します。

page.php

固定ページのファイル



っとした説明です。

- ・固定ページ同士で階層を作る事ができます。
- ・固定ページごとに適用するデザインテンプレートを切り替える事ができます。
- ・カスタムメニューを使って固定ページへのリンクを表示させる事ができます。
- ・投稿記事のように、ページにカテゴリやタグの設定をする事ができません。
- ・固定ページのコンテンツ内にPHPのコードを書いても動きません。

検索ワード

wordpress 固定ページ

固定ページのカスタムテンプレート

WordPressの固定ページごとに、レイアウトやデザインを変えたくないですか。

たとえば、固定ページごとにサイドバーのデザインを変更し、それぞれのページに合った広告やナビゲーションを配置するために。

それを可能にするのが『カスタムテンプレート』

固定ページ限定なので、ご注意ください。

以下の記述を加えるだけ

```
<?php
/*
Template Name: gallery
*/
?>
```



っとした説明です。

- ・テンプレートとなるPHPファイル(〇〇.php)を作成する。

【注意】

ファイル名は、WordPressで既に存在するテンプレート名・テンプレートパーツ名以外であれば、何でもOK。

検索ワード

wordpress カスタムテンプレート

条件分岐タグ

条件分岐タグは「PHP」というプログラミング言語により書きます。

メインページ	is_home()
フロントページ	is_front_page()
管理画面	is_admin()
投稿ページ	is_single()
固定ページ	is_page()
カテゴリーページ	is_category()
タグページ	is_tag()
日付別ページ	is_date()
アーカイブページ	is_archive()
検索結果ページ	is_search()
404 Not Found ページ	is_404()

have_postsの場合

```
<?php if(have_posts()): while(have_posts()): the_post();?>
<h1><?php the_title(); ?></h1>
<?php the_content(); ?>
<?php endwhile; endif; ?>
```

こちらのループ処理は、「記事があればタイトルを、h1タグで囲み、次に本文を表示させるという処理を、ある記事の数だけ繰り返して行ってください」という意味になります。

if文を使う

例：PHPのif文で書く

```
<?php if (条件) : ?>
```

条件に当てはまるときに表示するものをココに書く

```
<?php else: ?>
```

条件に当てはまらないときに表示するものをココに書く

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

例：elseifで細かな条件分岐ができる

```
<?php if ( is_single() ) : ?>
```

①記事ページで表示するものをココに書く

```
<?php elseif ( is_page() ) : ?>
```

②固定ページで表示するものをココに

```
<?php else: ?>
```

①と②のどちらにも当てはまらないときに表示するものをココに

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



っとした説明です。

- ・1つめのifに当てはまらなかったときに、2つめのelseifの条件を判定してくれるようになります。<?php elseif (条件) : ?>はいくつ繋げてOK !
- ・ifとelseifの後にはコロン(:) ※if() :とelseif() :というように書きます。 ;と書くとエラーになります。
- ・endifの後にはセミコロン(;) ※endif ;というように書きましょう。
- ・括弧の数に注意 ※また「括弧を書き忘れていないか」にも注意しましょう。(と)の数は一致するはずです。
- ・全角で書かない ; : ?phpなどは必ず半角英数字で書くようにしましょう。

検索ワード

wordpress if文

ループ処理のタグと合わせる

while → 処理を繰り返す構文

have_posts() → 次の記事があるか調べる

the_post() → 次の記事を取得する

投稿ページの記述例

```
<?php if(is_single()): while(have_posts()): the_post();?>
<?php the_content(); ?>
<?php endwhile; endif; ?>
```



※書いていることは同じです。

```
<?php if(is_single()): ?>
    <?php while(have_posts()): ?>
        <?php the_post();?>
        <?php the_content(); ?>
    <?php endwhile; ?>
<?php endif; ?>
```



っとした説明です。

while文は、for文と同様のループ処理を行うためのPHP構文で、条件が真の間だけ与えられた文の実行を繰り返すというループを記述するための文法。

- whileはendwhileで終わる
- 回数が決まっている時にはfor文で回数が未定の場合は while文を使うのが便利
- 永遠に処理を繰り返さないための処理が必要

検索ワード

wordpress While文

他のページに投稿を表示させる

query_postsの使い方

「query_posts」はループ処理の前に、表示させたい記事の条件を一行追加するだけで、記事件数や特定カテゴリーなどを抽出して一覧表示してくれます。

```
<?php query_posts('showposts=5&cat=0'); while(have_posts()) : the_post(); ?>  
<a href="<?php the_permalink(); ?>" title="<?php the_title(); ?>"><?php the_title(); ?></a>  
<?php endwhile; ?>
```

showposts=5&cat=0

は、「showposts=5」は表示件数を5件「cat=0」でカテゴリーを指定できます。

※ 0は全てのカテゴリーの意味を指します。

アイキャッチ画像の表示方法

WordPressの投稿記事では「アイキャッチ」と呼ばれるサムネイル画像を設定することができます。

任意の場所で以下のコードを記述

```
<?php the_post_thumbnail(); ?>
```

サイズ別で指定する場合

```
<?php the_post_thumbnail('thumbnail'); ?> //サムネイルのサイズ  
<?php the_post_thumbnail('medium'); ?> //中サイズ  
<?php the_post_thumbnail('large'); ?> //大サイズ  
<?php the_post_thumbnail('full'); ?> //フルサイズ
```

functions.php

```
add_theme_support('post-thumbnails');
```

CSSのクラス名を入れたい場合

```
<?php the_post_thumbnail('thumbnail', 'class=sample'); ?>
```

検索ワード

wordpressアイキャッチ画像

ウィジェットの活用について

sidebar.php

```
<?php dynamic_sidebar(sidebar); ?>
```

表示させたいパーツに上記のコードを記入

「functions.php」ウィジェット用のコードを記入
管理画面に「ウィジェット」が出てくるので
これで使用可能になる

「ウィジェット」で

- ・カレンダー
- ・アーカイブ
- ・検索機能

が実装できるので必要なファイルと設定が増えます。

functions.php

+

```
//ウィジェットの設置(1つ目)
if ( function_exists('register_sidebar'))
register_sidebar(array(
    'name'=>'sidebar',
    'id' => 'sidebar',
    'before_widget'=>'<div>',
    'after_widget'=>'</div>',
    'before_title' => '<h3>',
    'after_title' => '</h3>')
```

検索ワード

wordpress ウィジェット

入門編まとめ

その他にも様々なWordpressの活用法があります。
ただし、これらを自在に操るには「HTML」「CSS」をしっかりと理解している必要があります。

中級編としては、
「検索機能」「ページ送り」「お問い合わせフォーム」「カスタムテンプレート」等々
様々な「プラグイン」の活用法といった事が更に出来ます。

また、上級編としては、
管理画面のカスタマイズ、「functions.php」の編集
サーバの連動やあれこれ。。。

というようなイメージです。

まずは「HTML」「CSS」をしっかりと身につけて実践を経験して、次のSTEPで「オリジナルテーマ」と
考えて頂ければと思います。