

21 オリジナルテーマの作成 4

11. フロントページ（トップページ）用のテンプレートファイルを作成

11-1. フロントページのテンプレート優先順位

フロント（トップ）ページのテンプレート優先順位は以下の通りです。

```
front-page.php > home.php > index.php
```

ここまでフロント（トップ）ページのテンプレートとして使用してきた index.php は、WordPress のテーマファイルでは必要最低限かつ、汎用的な位置づけのテンプレートファイルです。フロント（トップ）ページのように、固有の要素を多く配置するページのテンプレートとしての使用は最優先の front-page.php を使用していくこととします。

11-2. front-page.php を作成する

まずは index.php を複製し、ファイル名を front-page.php に変更します。

その上で index.php の記載内容は削除します。

これでトップページは front-page.php を読み込んで表示される状態となりました。

12. 投稿一覧（インデックス）ページをスラッグ名 news として表示する

12-1. 最新投稿のテンプレート優先順位

最新投稿を表示する際のテンプレートの優先順位はフロント（トップ）ページのテンプレート優先順位と同様になります。

12-2. 投稿インデックスページ用のテンプレートを作成

テーマフォルダ内の archive.php を複製して、優先順位次点の home.php という名前に変更します。

12-3. トップページ・投稿インデックス用の固定ページを作成する。

管理画面 [固定ページ] > [新規固定ページを追加] から、

ページ名：「トップページ」、ページスラッグ（パーマリンク）：任意（ここでは top とします）

ページ名：「お知らせ」、ページスラッグ（パーマリンク）：news

の 2 つの固定ページを作成します。

12-4. 表示設定を変更

管理画面 [設定] > [表示設定] から、「ホームページの表示」のチェックを「固定ページ (以下で選択)」に変更します。

「ホームページ：」と「投稿ページ：」のプルダウンからそれぞれ 12-3 で作成した固定ページを選択して更新を保存します。

13. フロントページにお知らせ一覧を読み込む

13-3. フロントページのお知らせ一覧部分を読み込む

前項でフロントページとアーカイブページの投稿一覧が同じ HTML なので投稿部分をパーツ化しました。このとき作成した「loop-post.php」を、archive.php と同様にフロントページのお知らせ一覧部分に読み込みます。

archive.php の 26 行目あたり `<?php if (have_posts()) : ?>~<?php endif; ?>` までの記述をコピーし、front-page.php にある投稿 3 件分の HTML と置き換えます。

(archive.php の記述の違いとしては、フロントページの `<ul class="newsList">` には、fadein というクラス名があるので忘れずに追記しましょう)

front-page.php 変更前

～省略～

```
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <ul class="newsList fadein">
      <li class="newsList__item">
        <a href="<?php echo esc_url( home_url()); ?>/news/detail.html" class="newsList__link"></a>
        <div class="newsList__itemInner">
          <div class="newsList__img">
            
          </div>
          <div class="newsList__cont">
            <h3 class="newsList__ttl"> タイトルが入ります </h3>
            <p class="newsList__txt"> いつも CAFE ASABEN をご利用いただ～省略～ [...]</p>
            <div class="newsList__data">
              <a href="<?php echo esc_url( home_url()); ?>/news/category.html" class="newsList__cat"> カテゴリー </a>
              <p class="newsList__time">2023 年 0 月 00 日 </p>
            </div>
          </div>
        </li>
      </ul>
      <div class="btn__base txt--center fadein">
        <a href="<?php echo esc_url( home_url()); ?>/news/"> お知らせをもっと見る </a>
      </div>
    </div>
  </section>
  ～省略～
```

front-page.php 変更後

～省略～

```
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <?php if ( have_posts() ) : ?>
      <ul class="newsList fadein">
        <?php
        while ( have_posts() ) :
          the_post();
        ?>
        <?php get_template_part( 'template-parts/loop', 'post' ); ?>
      <?php endwhile; ?>
    </ul>
    <?php endif; ?>
    <div class="btn__base txt--center fadein">
      <a href="<?php echo esc_url( home_url() ); ?>/news/"> お知らせをもっと見る </a>
    </div>
  </div>
</section>
～省略～
```

トップページを表示してみましょう。現状はお知らせが全建表示されている状態なので、これを 3 件表示に設定しなおしていきます。

WordPress が投稿を表示するしくみ

WordPress では URL によるリクエストで取得するページを指定し、その URL の情報に基づいてウェブページを表示しています。この「URL による出力ページのリクエスト」のことを「メインクエリー」と呼びます。クエリーとは、「どのような条件でページの表示 (ループの実行) を行うかという問い合わせ」と考えます。

メインクエリーは URL の情報に基づいた全ての情報をリクエストします。

front-page.php に archive.php と同じ基本形のループを記述したため、フロントページでは最新の投稿一覧が表示されました。この基本形のループのことを、「メインループ」と呼ぶこともあります。

サブクエリー・サブループ

メインクエリーの表示条件は WordPress 内部にあらかじめ用意されているため、single.php や page.php、archive.php などのテンプレートではデータ取得の条件を記述する必要がありません。フロントページのお知らせ一覧のように「最新の投稿 3 件だけ」といった条件で出力したい場合は、メインクエリーとは別に独自のクエリーを作成して呼び出します。この独自に作成するクエリーを「サブクエリー」と呼び、サブクエリーで取得した情報によって出力するループのことを「サブループ」といいます。

サブループの基本的な記述

```
<?php
    $my_query = new WP_Query( 引数 );
    if ( $my_query->have_posts() ) :
?>
<?php
    while ( $my_query->have_posts() ) :
        $my_query->the_post();
?>
    // 表示する内容
<?php
    endwhile;
    wp_reset_postdata();
?>
<?php endif; ?>
```

取得する投稿の条件を new WP_Query() の引数に連想配列で定義します。WP_Query() が取得条件を WordPress にリクエストすると、返ってきた値が \$my_query という変数に格納されます。

\$my_query に投稿が存在すれば投稿の出力に進む

投稿がある間は処理を続ける

メインループで表示が行われているところに new WP_Query() で独自クエリーを定義すると、メインループの投稿データがサブループに差し替えられます。このままではページの表示に影響が出てしまうため、サブループの処理が終わった時点でメインループの状態に戻さなくてはなりません。そこで「wp_reset_postdata()」関数を記述し、サブループの投稿データをメインループの状態にリセットします。「WP_Query() と wp_reset_postdata() は必ずセットで使用します。

13-4. サブループを記述する

front-page.php にサブループを記述します。

front-page.php 変更前

～省略～

```
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <?php if ( have_posts() ) : ?>
      <ul class="newsList fadein">
        <?php
          while ( have_posts() ) :
            the_post();
            ?>
            <?php get_template_part( 'template-parts/loop', 'post' ); ?>
          <?php endwhile; ?>
        </ul>
      <?php endif; ?>
```

～省略～

front-page.php 変更後

～省略～

```
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <?php
      $asabenNews_query = new WP_Query( $asaben_args );
      if ( $asabenNews_query->have_posts() ) :
        ?>
        <ul class="newsList fadein">
          <?php
            while ( $asabenNews_query->have_posts() ) :
              $asabenNews_query->the_post();
              ?>
              <?php get_template_part( 'template-parts/loop', 'post' ); ?>
            <?php
              endwhile;
              wp_reset_postdata();
            ?>
          </ul>
        <?php endif; ?>
```

～省略～

13-5. 引数で条件を指定する

WP_Query() の引数は数多くあり、複数の条件を組み合わせることができます。フロントページのお知らせ一覧は、「投稿記事」「最新の投稿 3 件」の条件で投稿を出力します。ここでは WP_Query() の括弧内ではなく、「\$asaben_args」という名前の変数に連想配列で引数を定義します。

front-page.php 変更前

```

～省略～
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <?php
      $asabenNews_query = new WP_Query( $asaben_args );
      if ( $asaben_query->have_posts() ) :
        ?>
        <ul class="newsList fadein">
          <?php
            while ( $asabenNews_query->have_posts() ) :
              $asabenNews_query->the_post();
            ?>
            <?php get_template_part( 'template-parts/loop', 'post' ); ?>
          <?php
            endwhile;
            wp_reset_postdata();
          ?>
        </ul>
      <?php endif; ?>
    </div>
  </div>
  <?php endif; ?>
～省略～

```

front-page.php 変更後

```

～省略～
<section class="section index__news">
  <div class="content content--small">
    <div class="ttl__base txt--center fadein">
      <h3 class="ttl__base--jp"> お知らせ </h3>
      <span class="ttl__base--en font--en">News</span>
    </div>
    <?php
      $asaben_args = array(
        'post_type' => 'post',      ← 投稿タイプの条件：投稿記事のみ
        'posts_per_page' => 3,     ← 表示件数の条件：3 件
      );
      $asabenNews_query = new WP_Query( $asaben_args );
      if ( $asaben_query->have_posts() ) :
        ?>
        <ul class="newsList fadein">
          <?php
            while ( $asabenNews_query->have_posts() ) :
              $asabenNews_query->the_post();
            ?>
            <?php get_template_part( 'template-parts/loop', 'post' ); ?>
          <?php
            endwhile;
            wp_reset_postdata();
          ?>
        </ul>
      <?php endif; ?>
    </div>
  </div>
  <?php endif; ?>
～省略～

```