

# WORDPRESS クイックリファレンス

テーマ、フック、クエリ、REST API、カスタム投稿タイプ

## テーマの基本

### 必須ファイル

**style.css** テーマメタデータヘッダーとスタイル  
**index.php** フォールバックテンプレート (必須)  
**functions.php** テーマ設定、フック、エンキュー  
**screenshot.png** テーマプレビュー (1200x900)

### style.css ヘッダー

```
/*  
Theme Name: My Theme  
Theme URI: https://example.com  
Version: 1.0  
Requires at least: 6.0  
*/
```

### functions.php のセットアップ

```
add_action('after_setup_theme', function() {  
    add_theme_support('title-tag');  
    add_theme_support('post-thumbnails');  
    add_theme_support('html5', 'search-form', 'gallery');  
    register_nav_menus(['primary' => 'Main Nav']);  
});
```

## テンプレート階層

### ページテンプレート

**front-page.php** 静的フロントページ  
**home.php** ブログ投稿インデックス  
**single.php** 単一の投稿  
**page.php** 単一のページ  
**archive.php** 投稿アーカイブ (日付、カテゴリ)  
**category.php** カテゴリアーカイブ  
**search.php** 検索結果  
**404.php** 見つかりません  
**index.php** 最終フォールバック

### テンプレートパーツ

```
get_header(); // header.php  
get_footer(); // footer.php  
get_sidebar(); // sidebar.php  
get_template_part('parts/card'); // parts/card.php  
get_template_part('parts/card', 'featured');
```

### 検索順序 (単一投稿)

- 1 single-{post\_type}-{slug}.php
- 2 single-{post\_type}.php
- 3 single.php
- 4 singular.php
- 5 index.php

## ループ

### 標準ループ

```
<?php if (have_posts()) : while (have_posts()) : the_post(); ?>  
<?php the_title(); ?></h2>  
<?php the_content(); ?>  
<?php endwhile; endif; ?>
```

### テンプレートタグ

**the\_title()** 投稿タイトルを出力  
**the\_content()** 投稿コンテンツを出力  
**the\_excerpt()** 投稿の抜粋を出力  
**the\_permalink()** 投稿 URL を出力  
**the\_post\_thumbnail()** アイキャッチ画像を出力  
**the\_date()** 公開日を出力  
**the\_author()** 著者名を出力  
**the\_category()** カテゴリリンクを出力

### ゲッター (出力せず返す)

```
$title = get_the_title();  
$url = get_permalink();  
$id = get_the_ID();  
$thumb = get_the_post_thumbnail_url(null, 'medium');
```

## カスタム投稿タイプ

### CPT の登録

```
add_action('init', function() {  
    register_post_type('product', [  
        'labels' => ['name' => 'Products', 'singular_name' => 'Product'],  
        'public' => true,  
        'has_archive' => true,  
        'supports' => ['title', 'editor', 'thumbnail'],  
    ]);  
});
```

### カスタムタクソノミー

```
register_taxonomy('brand', 'product', [  
    'labels' => ['name' => 'Brands'],  
    'hierarchical' => true,  
    'rewrite' => ['slug' => 'brand'],  
]);
```

### supports オプション

**title** 投稿タイトルフィールド  
**editor** コンテンツエディター  
**thumbnail** アイキャッチ画像  
**excerpt** 抜粋フィールド  
**custom-fields** カスタムフィールドのメタボックス  
**page-attributes** メニュー順序、親  
**revisions** リビジョン履歴

## フックとフィルター

### アクション (何かを実行)

```
// Register a hook  
add_action('wp_head', 'my_analytics', 20);  
function my_analytics() {  
    echo '<script>/* tracking */</script>';  
}
```

### フィルター (データを変更)

```
// Modify the excerpt length  
add_filter('excerpt_length', fn() => 30);  
// Append text to every post  
add_filter('the_content', function($c) {  
    return is_single() ? $c . '<p>Thanks for reading!</p>' : $c;  
});
```

### 主要フック

**init** WP 初期化済み。ここで CPT を登録  
**wp\_enqueue\_scripts** CSS/JS をエンキュー  
**wp\_head** <head> へ出力  
**wp\_footer** </body> の前に出力  
**save\_post** 投稿が保存された後  
**pre\_get\_posts** 実行前にメタクエリを変更  
**the\_content** 投稿コンテンツ出力をフィルター  
**the\_title** 投稿タイトル出力をフィルター

## WP Query

### カスタムクエリ

```
$q = new WP_Query([  
    'post_type' => 'product',  
    'posts_per_page' => 10,  
    'order' => 'date',  
    'order' => 'DESC',  
]);  
while ($q->have_posts()) : $q->the_post();  
    the_title('<h3>'; </h3>);  
endwhile; wp_reset_postdata();
```

### 一般的なパラメーター

**post\_type** post, page, またはカスタムタイプ  
**posts\_per\_page** 結果数 (-1 ですべて)  
**category\_name** カテゴリスラッグでフィルター  
**tag** タグスラッグでフィルター  
**s** 検索キーワード  
**meta\_key / meta\_value** カスタムフィールドでフィルター  
**tax\_query** 高度なタクソノミーフィルタリング  
**date\_query** 日付範囲でフィルター

### メタクエリの変更

```
add_action('pre_get_posts', function($q) {  
    if (!is_admin() && $q->is_main_query() && is_home()) {  
        $q->set('posts_per_page', 5);  
    }  
});
```

## REST API

### 組み込みエンドポイント

**/wp-json/wp/v2/posts** 投稿の一覧取得 / 作成  
**/wp-json/wp/v2/posts/{id}** 投稿の取得 / 更新 / 削除  
**/wp-json/wp/v2/pages** ページエンドポイント  
**/wp-json/wp/v2/categories** カテゴリエンドポイント  
**/wp-json/wp/v2/media** メディアエンドポイント  
**/wp-json/wp/v2/users** ユーザーエンドポイント

### カスタムエンドポイント

```
add_action('rest_api_init', function() {  
    register_rest_route('myplugin/v1', '/items', [  
        'methods' => 'GET',  
        'callback' => fn($req) => new WP_REST_Response(  
            get_posts(['post_type' => 'item', 'numberposts' => 20])  
        ),  
        'permission_callback' => '__return_true',  
    ]);  
});
```

### JavaScript からの取得

```
const res = await fetch('/wp-json/wp/v2/posts?per_page=5');  
const posts = await res.json();  
posts.forEach(p => console.log(p.title.rendered));
```

## 関数

### 投稿関数

**get\_posts(\$args)** 投稿の記列を取得  
**wp\_insert\_post(\$arr)** 投稿を作成または更新  
**wp\_delete\_post(\$id)** 投稿をゴミ箱または完全削除

**get\_post\_meta(\$id, \$key)** カスタムフィールドの値を取得  
**update\_post\_meta(\$id, \$k, \$v)** カスタムフィールドの値を設定

### ユーザーと認証

**wp\_get\_current\_user()** 現在ログイン中のユーザー  
**is\_user\_logged\_in()** ログイン状態を確認  
**current\_user\_can(\$cap)** ユーザビリティを確認  
**wp\_create\_nonce(\$action)** セキュリティ nonce を生成  
**wp\_verify\_nonce(\$nonce, \$a)** nonce を検証

### ユーティリティ関数

**esc\_html(\$str)** HTML コンテキスト用にエスケープ  
**esc\_attr(\$str)** 属性コンテキスト用にエスケープ  
**esc\_url(\$url)** URL をサニタイズ  
**wp\_kses\_post(\$html)** 安全な投稿 HTML のみ許可  
**sanitize\_text\_field(\$s)** タグと余分な空白を削除  
**absint(\$val)** 絶対値の整数

## エンキュー

### スタイルとスクリプトのエンキュー

```
add_action('wp_enqueue_scripts', function() {  
    wp_enqueue_style('theme-style', get_stylesheet_uri(), [], '1.0');  
    wp_enqueue_script('theme-js', get_theme_file_uri('/js/app.js'),  
        ['jquery'], '1.0', true);  
});
```

## エンキュー関数

**wp\_enqueue\_style(\$h, \$src)** CSS を登録してエンキュー  
**wp\_enqueue\_script(\$h, \$src)** JS を登録してエンキュー  
**wp\_localize\_script(\$h, \$n, \$d)** PHP データを JS に渡す  
**wp\_dequeue\_style(\$handle)** エンキュー済みスタイルを削除  
**wp\_dequeue\_script(\$handle)** エンキュー済みスクリプトを削除

## JavaScript にデータを渡す

```
wp_enqueue_script('my-app', get_theme_file_uri('/js/app.js'));  
wp_localize_script('my-app', 'myData', {  
    'ajaxURL' => admin_url('admin-ajax.php'),  
    'nonce' => wp_create_nonce('my_action'),  
});  
// JS: console.log(myData.ajaxURL)
```

## よく使うパターン

### カスタムページテンプレート

```
<?php  
/* Template Name: Full Width */  
get_header(); ?>  
<main class="full-width">  
<?php the_content(); ?>  
</main>  
<?php get_footer(); ?>
```

### AJAX ハンドラー

```
add_action('wp_ajax_my_action', 'handle_my_action');  
add_action('wp_ajax_nopriv_my_action', 'handle_my_action');  
function handle_my_action() {  
    check_ajax_referer('my_action', 'nonce');  
    wp_send_json_success(['msg' => 'OK!']);  
}
```

### ショートコード

```
add_shortcode('greeting', function($atts) {  
    $a = shortcode_atts(['name' => 'World'], $atts);  
    return '<p>Hello, ' . esc_html($a['name']) . '!</p>';  
});  
// Usage: [greeting name="Zhi"]
```