

WORDPRESS 빠른 참조

테마, 훅, 쿼리, REST API, 커스텀 포스트 타입

테마 기본

필수 파일

- style.css** : 테마 메타데이터 헤더 + 스타일
- index.php** : 플백 템플릿 (필수)
- functions.php** : 테마 설정, 훅, 인큐
- screenshot.png** : 테마 미리보기 (1200x900)

style.css 헤더

```

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

```

functions.php 설정

```

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']);
});

```

템플릿 개념

페이지 템플릿

- front-page.php** : 정적 프론트 페이지
- home.php** : 블로그 포스트 인덱스
- single.php** : 단일 포스트
- page.php** : 단일 페이지
- archive.php** : 포스트 아카이브 (날짜, 카테고리)
- category.php** : 카테고리 아카이브
- search.php** : 검색 결과
- 404.php** : 찾을 수 없음
- index.php** : 최종 플백

템플릿 파트

```

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');

```

조회 순서 (단일 포스트)

- 1 single-{post_type}-{slug}.php
- 2 single-{post_type}.php
- 3 single.php
- 4 singular.php
- 5 index.php

루프

표준 루프

```

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

```

템플릿 태그

- the_title()** : 포스트 제목 출력
- the_content()** : 포스트 콘텐츠 출력
- the_excerpt()** : 포스트 발췌문 출력
- the_permalink()** : 포스트 URL 출력
- the_post_thumbnail()** : 특성 이미지 출력
- the_date()** : 게시물 출력
- the_author()** : 작성자 이름 출력
- the_category()** : 카테고리 링크 출력

게터 (출력 없이 반환)

```

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

```

커스텀 포스트 타입

CPT 등록

```

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

```

커스텀 분류

```

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

```

supports 옵션

- title** : 포스트 제목 필드
- editor** : 콘텐츠 에디터
- thumbnail** : 특성 이미지
- excerpt** : 발췌문 필드
- custom-fields** : 커스텀 필드 메타 박스
- page-attributes** : 메뉴 순서, 부모
- revisions** : 개정 이력

훅 및 필터

액션 (무언가 실행)

```

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

```

필터 (데이터 수정)

```

// 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;
});

```

주요 훅

- init** : WP 초기화됨, 여기서 CPT 등록
- wp_enqueue_scripts** : CSS/JS 인큐
- wp_head** : <head>에 출력
- wp_footer** : </body> 앞에 출력
- save_post** : 포스트 저장 후
- pre_get_posts** : 실행 전 메인 쿼리 수정
- the_content** : 포스트 콘텐츠 출력 필터
- the_title** : 포스트 제목 출력 필터

WP Query

커스텀 쿼리

```

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

```

일반적인 파라미터

- post_type** : post, page, 또는 커스텀 타입
- posts_per_page** : 결과 수 (-1이면 전체)
- category_name** : 카테고리 슬러그로 필터
- tag** : 태그 슬러그로 필터
- s** : 검색 키워드
- meta_key / meta_value** : 커스텀 필드로 필터
- tax_query** : 고급 분류 필터링
- date_query** : 날짜 범위로 필터

메인 쿼리 수정

```

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

```

REST API

내장 엔드포인트

- /wp-json/wp/v2/posts** : 포스트 목록 / 생성
- /wp-json/wp/v2/posts/{id}** : 포스트 조회 / 수정 / 삭제
- /wp-json/wp/v2/pages** : 페이지 엔드포인트
- /wp-json/wp/v2/categories** : 카테고리 엔드포인트
- /wp-json/wp/v2/media** : 미디어 라이브러리
- /wp-json/wp/v2/users** : 사용자 엔드포인트

커스텀 엔드포인트

```

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',
    ]);
});

```

JavaScript에서 가져오기

```

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));

```

함수

포스트 함수

- get_posts(\$args)** : 포스트 배열 가져오기
- wp_insert_post(\$arr)** : 포스트 생성 또는 업데이트
- wp_delete_post(\$id)** : 포스트를 삭제 또는 삭제
- get_post_meta(\$id, \$key)** : 커스텀 필드 값 가져오기
- update_post_meta(\$id, \$k, \$v)** : 커스텀 필드 값 설정

사용자 및 인증

- wp_get_current_user()** : 현재 로그인 사용자
- is_user_logged_in()** : 로그인 상태 확인
- current_user_can(\$cap)** : 권한 확인
- wp_create_nonce(\$action)** : 보안 nonce 생성
- wp_verify_nonce(\$nonce, \$a)** : nonce 검증

유틸리티 함수

- esc_html(\$str)** : HTML 콘텐츠용 이스케이프
- esc_attr(\$str)** : 속성 콘텐츠용 이스케이프
- esc_url(\$url)** : URL 새너타이즈
- wp_kses_post(\$html)** : 안전함 포스트 HTML만 허용
- sanitize_text_field(\$s)** : 태그 및 여분의 공백 제거
- absint(\$val)** : 절대 정수

인큐

스타일 및 스크립트 인큐

```

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

```

인큐 함수

- wp_enqueue_style(\$h, \$src)** : CSS 등록 및 인큐
- wp_enqueue_script(\$h, \$src)** : JS 등록 및 인큐

- wp_localize_script(\$h, \$n, \$d)** : PHP 데이터를 JS에 전달
- wp_dequeue_style(\$handle)** : 인큐된 스타일 제거
- wp_dequeue_script(\$handle)** : 인큐된 스크립트 제거

JavaScript에 데이터 전달

```

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)

```

자주 쓰는 패턴

커스텀 페이지 템플릿

```

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

```

AJAX 핸들러

```

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!']);
}

```

숏코드

```

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

```