

Referencia Rápida de WordPress

Temas, hooks, consultas, REST API, tipos de entrada personalizados

Fundamentos del Tema

Archivos Requeridos

style.css	Encabezado de metadatos del tema y estilos
index.php	Plantilla de respaldo (obligatoria)
functions.php	Configuración del tema, hooks, enqueue
screenshot.png	Vista previa del tema (1200x900)

Encabezado de style.css

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

Configuración de functions.php

```
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']);
});
```

Jerarquía de Plantillas

Plantillas de Página

front-page.php	Página de inicio estática
home.php	Índice de entradas del blog
single.php	Entrada individual
page.php	Página individual
archive.php	Archivos de entradas (fecha, categoría)
category.php	Archivo de categoría
search.php	Resultados de búsqueda
404.php	No encontrado
index.php	Respaldo final

Partes de Plantilla

```
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');
```

Orden de Búsqueda (Entrada Individual)

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

El Loop

Loop Estándar

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

Etiquetas de Plantilla

the_title()	Imprime el título de la entrada
the_content()	Imprime el contenido de la entrada
the_excerpt()	Imprime el extracto de la entrada
the_permalink()	Imprime la URL de la entrada
the_post_thumbnail()	Imprime la imagen destacada
the_date()	Imprime la fecha de publicación
the_author()	Imprime el nombre del autor
the_category()	Imprime los enlaces de categoría

Getters (Devuelven, No Imprimen)

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

Tipos de Entrada Personalizados

Registrar CPT

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

Taxonomía Personalizada

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

Opciones de Supports

title	Campo de título de la entrada
editor	Editor de contenido
thumbnail	Imagen destacada
excerpt	Campo de extracto
custom-fields	Meta box para campos personalizados
page-attributes	Orden de menú, padre
revisions	Historial de revisiones

Hooks y Filtros

Actions (Hacer Algo)

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

Filters (Modificar Datos)

```
// Modificar la longitud del extracto
add_filter('excerpt_length', fn() => 30);
// Añadir texto a cada entrada
add_filter('the_content', function($c) {
    return is_single() ? $c . '<p>Thanks for reading!</p>' : $c;
});
```

Hooks Clave

init	WP inicializado; registrar CPTs aquí
wp_enqueue_scripts	Encolar CSS/JS
wp_head	Salida en <head>
wp_footer	Salida antes de </body>
save_post	Después de guardar una entrada
pre_get_posts	Modificar la consulta principal antes de ejecutar
the_content	Filtrar salida del contenido de la entrada
the_title	Filtrar salida del título de la entrada

WP_Query

Consulta Personalizada

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

Parámetros Comunes

post_type	post, page o tipo personalizado
posts_per_page	Número de resultados (-1 para todos)
category_name	Filtrar por slug de categoría
tag	Filtrar por slug de etiqueta
s	Palabra clave de búsqueda
meta_key / meta_value	Filtrar por campo personalizado
tax_query	Filtrado avanzado de taxonomías
date_query	Filtrar por rango de fechas

Modificar Consulta Principal

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

REST API

Endpoints Integrados

/wp-json/wp/v2/posts	Listar / crear entradas
/wp-json/wp/v2/posts/{id}	Obtener / actualizar / eliminar entrada
/wp-json/wp/v2/pages	Endpoint de páginas
/wp-json/wp/v2/categories	Endpoint de categorías
/wp-json/wp/v2/media	Biblioteca de medios
/wp-json/wp/v2/users	Endpoint de usuarios

Endpoint Personalizado

```
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',
    ]);
});
```

Obtener desde JavaScript

```
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));
```

Referencia Rápida de WordPress

Funciones

Funciones de Entrada

<code>get_posts(\$args)</code>	Recuperar arreglo de entradas
<code>wp_insert_post(\$arr)</code>	Crear o actualizar una entrada
<code>wp_delete_post(\$id)</code>	Enviar a papelera o eliminar una entrada
<code>get_post_meta(\$id, \$key)</code>	Obtener valor de campo personalizado
<code>update_post_meta(\$id, \$k, \$v)</code>	Establecer valor de campo personalizado

Usuario y Autenticación

<code>wp_get_current_user()</code>	Usuario actual conectado
<code>is_user_logged_in()</code>	Verificar estado de sesión
<code>current_user_can(\$cap)</code>	Verificar capacidad
<code>wp_create_nonce(\$action)</code>	Generar nonce de seguridad
<code>wp_verify_nonce(\$nonce, \$a)</code>	Verificar nonce

Funciones de Utilidad

<code>esc_html(\$str)</code>	Escapar para contexto HTML
<code>esc_attr(\$str)</code>	Escapar para contexto de atributo
<code>esc_url(\$url)</code>	Sanear URL
<code>wp_kses_post(\$html)</code>	Permitir solo HTML seguro para entradas
<code>sanitize_text_field(\$s)</code>	Eliminar etiquetas y espacios en blanco extra
<code>absint(\$val)</code>	Entero absoluto

Enqueue

Encolar Estilos y Scripts

```
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);
});
```

Funciones de Enqueue

<code>wp_enqueue_style(\$h, \$src)</code>	Registrar + encolar CSS
<code>wp_enqueue_script(\$h, \$src)</code>	Registrar + encolar JS
<code>wp_localize_script(\$h, \$n, \$d)</code>	Pasar datos PHP a JS
<code>wp_dequeue_style(\$handle)</code>	Eliminar estilo encolado
<code>wp_dequeue_script(\$handle)</code>	Eliminar script encolado

Pasar Datos a JavaScript

```
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)
```

Patrones Comunes

Plantilla de Página Personalizada

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

Manejador AJAX

```
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']);
}
```

Shortcode

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