

60 نمونه سوال عملی وردپرس با جواب

در این مجموعه، 60 سوال و پاسخ عملی برای کار با وردپرس آماده شده است. این سوالات طیف وسیعی از موضوعات، از مبانی اولیه تا مباحث پیشرفته توسعه و امنیت را پوشش می‌دهند و برای سنجش مهارت و دانش افراد در سطوح مختلف مناسب هستند.

بخش اول: مبانی و مفاهیم کلیدی وردپرس

۱. تفاوت اصلی بین `wordpress.org` و `wordpress.com` چیست؟

این دو، دو روی متفاوت از سکه وردپرس هستند. `wordpress.org` نسخه متن-باز و رایگان وردپرس است که شما می‌توانید آن را روی هاست شخصی خود نصب کنید. در این حالت، کنترل کامل بر روی همه چیز، از جمله قالب‌ها، افزونه‌ها و کدهای سایت خود دارید. اما مسئولیت نگهداری، امنیت و پشتیبان‌گیری نیز با خود شماست.

در مقابل، `wordpress.com` یک سرویس میزبانی مدیریت‌شده است که توسط خود شرکت Automattic (خالقان وردپرس) ارائه می‌شود. در این سرویس، شما نیازی به خرید هاست جداگانه ندارید و مسائل فنی مانند نصب و نگهداری بر عهده سرویس‌دهنده است. با این حال، در پلن‌های رایگان و ارزان‌تر، محدودیت‌هایی در زمینه نصب افزونه‌ها، استفاده از قالب‌های سفارشی و ویرایش کدها وجود دارد.

به طور خلاصه، `wordpress.org` برای کسانی مناسب است که به دنبال کنترل کامل و انعطاف‌پذیری هستند، در حالی که `wordpress.com` برای کاربرانی که راحتی و سادگی را ترجیح می‌دهند، گزینه بهتری است.

۲. پست (Post) و برگه (Page) در وردپرس چه تفاوتی با هم دارند؟

گرچه هر دو برای ایجاد محتوا استفاده می‌شوند، اما کاربرد متفاوتی دارند. پست‌ها برای محتوای پویا و زمان‌محور مانند مقالات وبلاگ، اخبار و اطلاعیه‌ها به کار می‌روند. آن‌ها معمولاً در یک ترتیب زمانی معکوس (جدیدترین در بالا) نمایش داده می‌شوند و می‌توانند دسته‌بندی و برچسب‌گذاری شوند.

برگه‌ها برای محتوای ثابت و ایستا مانند «درباره ما»، «تماس با ما» یا «خدمات» استفاده می‌شوند. این صفحات معمولاً تاریخ انقضا ندارند و در ساختار سلسله‌مراتبی (مانند زیرمجموعه) قابل سازماندهی هستند.

۳. نقش «قالب فرزند» (Child Theme) چیست و چرا استفاده از آن توصیه می‌شود؟

قالب فرزند، یک قالب است که تمام ویژگی‌ها و استایل‌های یک قالب دیگر (قالب والد) را به ارث می‌برد. استفاده از قالب فرزند به شما این امکان را می‌دهد که تغییرات و سفارشی‌سازی‌های خود را در فایل‌های جداگانه‌ای اعمال کنید، بدون اینکه نیازی به دستکاری فایل‌های اصلی قالب والد داشته باشید.

مزیت اصلی این کار این است که وقتی قالب والد به‌روزرسانی می‌شود، تغییرات شما از بین نمی‌روند. اگر شما مستقیماً فایل‌های قالب اصلی را ویرایش کنید، با هر آپدیت، تمام زحمات شما پاک خواهد شد. بنابراین، استفاده از قالب فرزند یک **روش استاندارد و حرفه‌ای** برای سفارشی‌سازی قالب‌ها در وردپرس است.

۴. هوک (Hook) در وردپرس به چه معناست و انواع اصلی آن کدامند؟

هوک‌ها یکی از قدرتمندترین ویژگی‌های وردپرس برای توسعه‌دهندگان هستند. آن‌ها نقاطی در کد وردپرس (هسته، قالب‌ها و افزونه‌ها) هستند که به شما اجازه می‌دهند تا بدون تغییر در فایل‌های اصلی، عملکرد یا خروجی وردپرس را تغییر دهید یا قابلیت جدیدی به آن اضافه کنید.

دو نوع اصلی هوک وجود دارد:

- **اکشن هوک (Action Hook):** به شما اجازه می‌دهد تا در یک نقطه مشخص از اجرای وردپرس، یک تابع یا عملیات خاص را اجرا کنید. برای مثال، می‌توانید با استفاده از اکشن `wp_head`، کدی را به بخش `<head>` سایت خود اضافه کنید.
- **فیلتر هوک (Filter Hook):** به شما امکان می‌دهد تا داده‌ها را قبل از نمایش یا ذخیره‌سازی، تغییر دهید. برای مثال، می‌توانید با فیلتر `the_content`، محتوای یک پست را قبل از نمایش به کاربر، ویرایش کنید.

۵. فایل `wp-config.php` چه نقشی دارد و شامل چه اطلاعات مهمی است؟

فایل `wp-config.php` یکی از حیاتی‌ترین فایل‌های نصب وردپرس است. این فایل به صورت پیش‌فرض در بسته دانلودی وردپرس وجود ندارد و در طی فرآیند نصب، بر اساس اطلاعاتی که شما وارد می‌کنید، ساخته می‌شود. این فایل به عنوان پل ارتباطی بین وردپرس و پایگاه داده عمل می‌کند.

مهم‌ترین اطلاعاتی که در این فایل ذخیره می‌شوند عبارتند از:

- اطلاعات اتصال به پایگاه داده (نام پایگاه داده، نام کاربری، رمز عبور و میزبان)
- کلیدهای امنیتی و نمک‌ها (Authentication Keys and Salts) برای افزایش امنیت
- پیشوند جداول پایگاه داده (`table_prefix$`)

- فعال‌سازی حالت دیباگ وردپرس (WP_DEBUG)

به دلیل حساسیت اطلاعات موجود در این فایل، محافظت از آن اهمیت بسیار بالایی دارد.

۶. شورت‌کد (Shortcode) چیست و چگونه می‌توان از آن استفاده کرد؟

شورت‌کدها قطعه‌کدهای کوتاهی هستند که در داخل براکت [] قرار می‌گیرند و به شما اجازه می‌دهند تا بدون نیاز به کدنویسی پیچیده، محتوای پویا یا قابلیت‌های خاصی را به پست‌ها، برگه‌ها و ویجت‌های خود اضافه کنید. برای مثال، یک افزونه گالری تصاویر ممکن است یک شورت‌کد مانند [my_gallery] ارائه دهد که با قرار دادن آن در ویرایشگر، گالری تصاویر نمایش داده می‌شود.

وردپرس به صورت پیش‌فرض چند شورت‌کد داخلی مانند [gallery] و [caption] دارد. افزونه‌ها و قالب‌های مختلف نیز می‌توانند شورت‌کدهای سفارشی خود را اضافه کنند. همچنین، توسعه‌دهندگان می‌توانند با استفاده از تابع add_shortcode()، شورت‌کدهای اختصاصی خود را بسازند.

۷. چگونه می‌توان امنیت یک سایت وردپرسی را افزایش داد؟ (حداقل ۵ مورد)

امنیت یک فرآیند مداوم است، نه یک اقدام یک‌باره. برخی از مهم‌ترین اقدامات برای افزایش امنیت وردپرس عبارتند از:

1. استفاده از رمزهای عبور قوی و منحصر به فرد: برای تمام حساب‌های کاربری، به ویژه مدیران، از رمزهای پیچیده استفاده کنید.
2. به‌روزرسانی منظم: همیشه هسته وردپرس، قالب‌ها و افزونه‌ها را به آخرین نسخه آپدیت کنید.
3. نصب یک افزونه امنیتی معتبر: افزونه‌هایی مانند Wordfence Security یا Sucuri Security می‌توانند قابلیت‌هایی مانند فایروال، اسکن بدافزار و محافظت از ورود را فراهم کنند.
4. محدود کردن تلاش‌های ناموفق برای ورود: این کار از حملات Brute Force جلوگیری می‌کند.
5. تهیه نسخه پشتیبان (Backup) به صورت منظم: در صورت بروز هرگونه مشکل، بتوانید سایت خود را به حالت قبل بازگردانید.
6. تغییر URL پیش‌فرض ورود به پیشخوان (wp-admin/): این کار ربات‌ها را برای پیدا کردن صفحه ورود شما سردرگم می‌کند.

۸. ویجت (Widget) در وردپرس چیست و در کجاها می‌توان از آن استفاده کرد؟

ویجت‌ها بلوک‌های کوچکی از محتوا هستند که می‌توان آن‌ها را در نواحی مشخصی از قالب سایت، که به آن‌ها «ناحیه ابزارک» یا Sidebar گفته می‌شود، قرار داد. این نواحی معمولاً در سایدبار، فوتر یا هدر قالب قرار دارند.

ویجت‌ها به شما اجازه می‌دهند تا بدون نیاز به کدنویسی، قابلیت‌های مختلفی مانند لیست آخرین پست‌ها، فرم جستجو، تقویم، ابر برچسب‌ها یا محتوای سفارشی را به سایت خود اضافه کنید. مدیریت ویجت‌ها از طریق بخش «نمایش > ابزارک‌ها» در پیشخوان وردپرس انجام می‌شود.

۹. پست تایپ سفارشی (Custom Post Type) چیست و چه زمانی به آن نیاز داریم؟

وردپرس به صورت پیش‌فرض دو نوع پست اصلی دارد: پست‌ها و برگه‌ها. اما گاهی اوقات شما نیاز به نوع محتوای دیگری دارید که ساختار متفاوتی دارد. برای مثال، یک سایت نمونه کارها ممکن است به نوع پستی به نام «نمونه کار» نیاز داشته باشد، یا یک سایت املاک به «ملک» و یک سایت فیلم به «فیلم».

پست تایپ سفارشی به شما اجازه می‌دهد تا این نوع محتواهای جدید را با فیلدها و ویژگی‌های خاص خودشان ایجاد کنید. این کار به سازماندهی بهتر محتوا و مدیریت آسان‌تر سایت کمک شایانی می‌کند. پست تایپ‌های سفارشی را می‌توان از طریق کدنویسی در فایل `functions.php` یا با استفاده از افزونه‌هایی مانند Custom Post Type UI ایجاد کرد.

۱۰. حلقه وردپرس (The Loop) چیست و چه کاربردی دارد؟

حلقه وردپرس، قطعه کد PHP است که برای نمایش پست‌ها در سایت شما استفاده می‌شود. وردپرس از این حلقه برای پردازش هر پستی که باید در صفحه فعلی نمایش داده شود، استفاده می‌کند. در داخل حلقه، شما به تمام اطلاعات مربوط به آن پست، مانند عنوان، محتوا، نویسنده، تاریخ و... دسترسی دارید و می‌توانید آن‌ها را با استفاده از توابع وردپرس (Template Tags) نمایش دهید.

ساختار اصلی حلقه به این صورت است:

```
if ( have_posts() ) {
    while ( have_posts() ) {
        the_post();
        // کدهای مربوط به نمایش محتوای پست
    }
}
```

این حلقه، قلب تپنده نمایش محتوا در قالب‌های وردپرس است.

۱۱. چگونه می‌توان یک افزونه را به صورت دستی (بدون استفاده از پیشخوان) نصب کرد؟

گاهی اوقات ممکن است به دلیل محدودیت‌های سرور یا دلایل دیگر، نتوانید یک افزونه را مستقیماً از پیشخوان وردپرس نصب کنید. در این حالت، می‌توانید آن را به صورت دستی نصب کنید:

1. ابتدا فایل فشرده (zip) افزونه را از منبع آن (مانند مخزن وردپرس) دانلود کنید.
2. فایل zip را در کامپیوتر خود از حالت فشرده خارج کنید. شما یک پوشه با نام افزونه خواهید داشت.
3. با استفاده از یک کلاینت FTP (مانند FileZilla) یا از طریق مدیریت فایل هاست خود، به پوشه `wp-content/plugins` در محل نصب وردپرس خود بروید.
4. پوشه افزونه را که در مرحله ۲ استخراج کردید، در این مسیر آپلود کنید.
5. در نهایت، به بخش «افزونه‌ها» در پیشخوان وردپرس خود بروید. افزونه جدید را در لیست مشاهده خواهید کرد. روی دکمه «فعال کردن» کلیک کنید.

۱۲. تفاوت بین دسته‌بندی (Category) و برچسب (Tag) چیست؟

هر دو برای سازماندهی پست‌ها استفاده می‌شوند، اما یک تفاوت کلیدی در ساختار و کاربرد آن‌ها وجود دارد.

دسته‌بندی‌ها برای گروه‌بندی گسترده و موضوعی پست‌های شما هستند. آن‌ها **سلسله‌مراتبی** هستند، یعنی شما می‌توانید زیرمجموعه‌هایی برای آن‌ها تعریف کنید (مانند یک دسته‌بندی «ورزشی» با زیرمجموعه‌های «فوتبال» و «والیبال»). هر پست معمولاً به یک یا دو دسته‌بندی تعلق دارد.

برچسب‌ها برای توصیف جزئیات خاص پست‌های شما به کار می‌روند. آن‌ها **سلسله‌مراتبی نیستند** و یک پست می‌تواند برچسب‌های زیادی داشته باشد. برچسب‌ها مانند کلمات کلیدی در یک مقاله عمل می‌کنند. برای مثال، یک پست در دسته‌بندی «فوتبال» می‌تواند برچسب‌های «لیگ برتر»، «رئال مادرید» و «کریستیانو رونالدو» را داشته باشد.

۱۳. چگونه می‌توان حالت خطایابی (Debug Mode) را در وردپرس فعال کرد؟

فعال کردن حالت خطایابی برای شناسایی و رفع مشکلات در سایت بسیار مفید است. برای این کار، باید فایل `wp-config.php` را ویرایش کنید.

در این فایل، خط زیر را پیدا کنید:

```
define( 'WP_DEBUG', false );
```

و مقدار `false` را به `true` تغییر دهید:

```
define( 'WP_DEBUG', true );
```

با این کار، تمام خطاها، هشدارها و اخطارهای PHP در صفحات سایت شما نمایش داده می‌شوند. برای یک سایت در حال کار (Production)، بهتر است که خطاها در یک فایل لاگ ذخیره شوند و به کاربران نمایش داده نشوند. برای این کار می‌توانید کدهای زیر را نیز به `wp-config.php` اضافه کنید:

```
define( 'WP_DEBUG_LOG', true );  
define( 'WP_DEBUG_DISPLAY', false );  
@ini_set( 'display_errors', 0 );
```

۱۴. فایل `functions.php` در یک قالب وردپرس چه کاربردی دارد؟

فایل `functions.php` مانند یک افزونه اختصاصی برای قالب شما عمل می‌کند. کدهایی که در این فایل قرار می‌دهید، فقط زمانی اجرا می‌شوند که آن قالب فعال باشد. این فایل برای اضافه کردن ویژگی‌ها و قابلیت‌های سفارشی به قالب شما استفاده می‌شود.

برخی از کاربردهای رایج این فایل عبارتند از:

- ثبت نواحی منو و سایدبار
- اضافه کردن پشتیبانی از ویژگی‌های وردپرس مانند تصویر شاخص، فرمت‌های پست و...
- فراخوانی فایل‌های CSS و JavaScript (Enqueueing)
- ایجاد شورت‌کدهای سفارشی
- تعریف توابع و هوک‌های اختصاصی برای قالب

۱۵. چگونه می‌توان از اجرای اسکریپت‌ها و استایل‌های یک افزونه در صفحات غیرضروری جلوگیری کرد؟

بسیاری از افزونه‌ها، فایل‌های CSS و JavaScript خود را در تمام صفحات سایت بارگذاری می‌کنند، حتی اگر فقط در یک صفحه خاص استفاده شوند. این کار می‌تواند سرعت سایت را کاهش دهد. برای جلوگیری از این موضوع، می‌توانید از هوک‌های `wp_enqueue_scripts` استفاده کنید.

شما باید شناسه (`handle`) اسکریپت یا استایل مورد نظر را پیدا کنید (معمولاً با بررسی سورس صفحه قابل مشاهده است) و سپس با استفاده از توابع `wp_dequeue_script` یا

wp_dequeue_style() ، آن‌ها را در صفحاتی که به آن‌ها نیاز ندارید، از صف خارج کنید. این کار معمولاً با استفاده از تگ‌های شرطی وردپرس (Conditional Tags) مانند is_page() یا is_single() انجام می‌شود.

برای مثال، کد زیر اسکریپت یک افزونه فرم تماس را فقط در صفحه «تماس با ما» بارگذاری می‌کند:

```
add_action( 'wp_enqueue_scripts', 'manage_plugin_assets', 99 );
function manage_plugin_assets() {
    // اگر در صفحه
    تماس با ما نیستیم
    if ( ! is_page( 'contact-us' ) ) {
        wp_dequeue_style( 'contact-form-style' );
        wp_dequeue_script( 'contact-form-script' );
    }
}
```

۱۶. REST API در وردپرس چیست و چه کاربردهایی دارد؟

REST API وردپرس یک رابط برنامه‌نویسی کاربردی است که به توسعه‌دهندگان اجازه می‌دهد تا از طریق درخواست‌های HTTP (مانند GET, POST, PUT, DELETE) با محتوای سایت وردپرسی تعامل داشته باشند. خروجی این API به صورت پیش‌فرض در فرمت JSON است که خواندن و استفاده از آن برای اپلیکیشن‌های مختلف بسیار ساده است.

کاربردهای REST API بسیار گسترده است، از جمله:

- **ساخت اپلیکیشن‌های موبایل (iOS و Android):** که محتوای خود را از سایت وردپرسی دریافت می‌کنند.
- **ایجاد وب‌اپلیکیشن‌های تک‌صفحه‌ای (Single-Page Applications):** با استفاده از فریم‌ورک‌های جاوااسکریپت مانند React یا Vue.js.
- **یکپارچه‌سازی با سرویس‌های دیگر:** ارسال یا دریافت داده بین سایت وردپرسی شما و سایر نرم‌افزارها.
- **اتوماسیون وظایف:** ایجاد اسکریپت‌هایی برای مدیریت محتوا به صورت خودکار.

۱۷. چگونه می‌توان اندازه آپلود فایل در وردپرس را افزایش داد؟

به صورت پیش‌فرض، وردپرس و هاستینگ شما ممکن است محدودیتی برای حداکثر حجم فایل‌های قابل آپلود (مانند تصاویر، ویدئوها یا فایل‌های فشرده) داشته باشند. برای افزایش این محدودیت، چندین راه وجود دارد:

1. **ویرایش فایل `php.ini`** : اگر به این فایل دسترسی دارید، می‌توانید مقادیر

`upload_max_filesize` و `post_max_size` را افزایش دهید.

2. **ویرایش فایل `htaccess`** : با افزودن کدهایی مانند `php_value upload_max_filesize`

64M می‌توان این محدودیت را تغییر داد.

3. **ویرایش فایل `functions.php`** : با افزودن کدهایی خاص به این فایل نیز می‌توان مقادیر را تغییر

داد.

4. **تماس با پشتیبانی هاست**: ساده‌ترین راه معمولاً این است که از شرکت هاستینگ خود بخواهید

این محدودیت را برای شما افزایش دهند.

۱۸. **WP_Query چیست و چه تفاوتی با `get_posts()` دارد؟**

`WP_Query` یک کلاس بسیار قدرتمند در وردپرس است که به شما اجازه می‌دهد تا کوئری‌های پیچیده‌ای را برای دریافت پست‌ها از پایگاه داده اجرا کنید. شما می‌توانید با تعیین پارامترهای مختلف، پست‌ها را بر اساس دسته‌بندی، برچسب، نویسنده، فیلدهای سفارشی و ده‌ها معیار دیگر فیلتر و دریافت کنید. `WP_Query` حلقه اصلی وردپرس را تغییر می‌دهد و برای ساخت حلقه‌های سفارشی (Custom Loops) استفاده می‌شود.

`get_posts()` در واقع یک تابع پوششی (wrapper) برای `WP_Query` است. استفاده از آن ساده‌تر است و یک آرایه از پست‌ها را برمی‌گرداند، اما تمام قدرت و انعطاف‌پذیری `WP_Query` را ندارد. `get_posts()` حلقه اصلی را تغییر نمی‌دهد و برای دریافت لیست‌های ساده‌ای از پست‌ها (مثلاً در یک ویجت) مناسب است.

به طور کلی، برای کوئری‌های پیچیده و ساخت حلقه‌های نمایش محتوا، `WP_Query` انتخاب بهتری است.

۱۹. **چگونه می‌توان یک صفحه ورود سفارشی برای وردپرس طراحی کرد؟**

صفحه ورود پیش‌فرض وردپرس (`wp-login.php`) ظاهر ساده‌ای دارد و لوگوی وردپرس را نمایش می‌دهد. برای برندسازی و ایجاد تجربه کاربری بهتر، می‌توانید این صفحه را سفارشی کنید. راه‌های مختلفی برای این کار وجود دارد:

- **استفاده از افزونه**: افزونه‌های زیادی مانند `Custom Login Page Customizer` یا `LoginPress` وجود دارند که به شما امکان می‌دهند تا به راحتی و بدون کدنویسی، ظاهر صفحه ورود را تغییر دهید (تغییر لوگو، پس‌زمینه، رنگ‌ها و...).

- **کدنویسی در `functions.php`**: شما می‌توانید با استفاده از هوک‌هایی مانند `login_enqueue_scripts` برای افزودن CSS سفارشی و هوک `login_headerurl` برای تغییر لینک لوگو، صفحه ورود را به دلخواه خود تغییر دهید.

۲۰. نقش فایل htaccess . در وردپرس چیست؟

فایل htaccess . یک فایل پیکربندی قدرتمند برای وب سرور آپاچی است. وردپرس از این فایل عمدتاً برای مدیریت پیوندهای یکتا (Permalinks) استفاده می‌کند. وقتی شما ساختار پیوندهای یکتای خود را از حالت پیش فرض تغییر می‌دهید، وردپرس قوانینی را در این فایل می‌نویسد تا درخواست‌های کاربران را به درستی به فایل index.php وردپرس هدایت کند.

علاوه بر این، از فایل htaccess . می‌توان برای مقاصد امنیتی و بهینه‌سازی نیز استفاده کرد، مانند:

- محافظت از فایل wp-config.php
- جلوگیری از Directory Listing
- ریدایرکت کردن آدرس‌ها
- فعال کردن فشرده‌سازی Gzip
- تنظیمات کش مرورگر

بخش دوم: توسعه قالب و افزونه

۲۱. برای ساخت یک قالب وردپرس از پایه، حداقل چه فایل‌هایی نیاز است؟

برای اینکه وردپرس یک قالب را شناسایی کند و بتوان آن را فعال کرد، حداقل به دو فایل در پوشه قالب نیاز دارید:

1. **style.css**: این فایل فقط برای استایل‌دهی نیست. وردپرس اطلاعات قالب (نام، نویسنده، نسخه و...) را از کامنت‌های موجود در هدر این فایل می‌خواند. بدون این فایل، قالب در لیست قالب‌ها نمایش داده نمی‌شود.

2. **index.php**: این فایل، فایل قالب پیش فرض است. اگر وردپرس نتواند فایل قالب مناسب‌تری برای یک صفحه پیدا کند (مثلاً **single.php** برای یک پست تکی یا **page.php** برای یک برگه)، به سراغ **index.php** می‌رود. این فایل شامل حلقه اصلی وردپرس برای نمایش پست‌هاست.

البته یک قالب کاربردی به فایل‌های بیشتری مانند **header.php** , **footer.php** , **sidebar.php** , **functions.php** و **single.php** نیاز خواهد داشت.

۲۲. تابع `wp_enqueue_script()` و `wp_enqueue_style()` چه کاری انجام می‌دهند و چرا استفاده از آن‌ها مهم است؟

این دو تابع، روش استاندارد و صحیح برای افزودن فایل‌های جاوااسکریپت و CSS به یک سایت وردپرسی هستند. به جای اینکه تگ‌های `<script>` و `<link>` را مستقیماً در **header.php** یا **footer.php**

قرار دهید، باید از این توابع در فایل `functions.php` (با استفاده از هوک `wp_enqueue_scripts`) استفاده کنید.

دلایل اهمیت این کار:

- **مدیریت وابستگی‌ها (Dependencies):** شما می‌توانید مشخص کنید که یک اسکریپت به اسکریپت دیگری (مانند jQuery) وابسته است. وردپرس تضمین می‌کند که اسکریپت وابسته، قبل از اسکریپت شما بارگذاری شود.
- **جلوگیری از بارگذاری تکراری:** اگر چندین افزونه یا قالب تلاش کنند یک اسکریپت یکسان (مثلاً یک کتابخانه خاص) را بارگذاری کنند، وردپرس فقط یک بار آن را بارگذاری می‌کند و از تداخل جلوگیری می‌کند.
- **انعطاف‌پذیری:** به سایر توسعه‌دهندگان اجازه می‌دهد تا در صورت نیاز، اسکریپت‌ها یا استایل‌های شما را با استفاده از توابع `wp_dequeue_script/style` حذف یا جایگزین کنند.

۲۳. چگونه می‌توان یک سایدبار یا ناحیه ابزارک سفارشی (Custom Widget Area) ثبت کرد؟

برای این کار باید دو مرحله را طی کنید:

1. **ثبت ناحیه ابزارک:** ابتدا باید با استفاده از تابع `register_sidebar()` در فایل `functions.php`، ناحیه ابزارک جدید خود را به وردپرس معرفی کنید. شما می‌توانید نام، شناسه و توضیحات آن را مشخص کنید.

```
function my_custom_sidebar() {
    register_sidebar( array(
        'name'          => 'سایدبار سفارشی من',
        'id'            => 'custom-sidebar',
        'description'  => 'این یک ناحیه ابزارک سفارشی است',
    ) );
}
add_action( 'widgets_init', 'my_custom_sidebar' );
```

2. **نمایش ناحیه ابزارک در قالب:** سپس، در هر فایل قالبی که می‌خواهید این ناحیه نمایش داده شود (مثلاً `sidebar.php` یا `footer.php`)، باید از تابع `dynamic_sidebar()` با شناسه‌ای که در مرحله قبل تعریف کردید، استفاده کنید.

```
if ( is_active_sidebar( 'custom-sidebar' ) ) {  
    dynamic_sidebar( 'custom-sidebar' );  
}
```

۲۴. فیلد سفارشی (Custom Field) چیست و چگونه می‌توان از آن برای افزودن اطلاعات اضافی به پست‌ها استفاده کرد؟

فیلدهای سفارشی (که به آن‌ها Post Meta نیز گفته می‌شود) به شما اجازه می‌دهند تا اطلاعات اضافی و ساختاریافته‌ای را به پست‌ها، برگه‌ها یا پست تایپ‌های سفارشی خود اضافه کنید. برای مثال، برای یک پست تایپ «فیلم»، می‌توانید فیلدهای سفارشی برای «کارگردان»، «سال تولید» و «امتیاز» ایجاد کنید.

وردپرس یک رابط کاربری ساده برای افزودن فیلدهای سفارشی در صفحه ویرایش پست ارائه می‌دهد. شما یک «نام» (Key) و یک «مقدار» (Value) برای هر فیلد وارد می‌کنید.

برای نمایش این اطلاعات در قالب، می‌توانید از تابع `get_post_meta()` استفاده کنید. این تابع مقدار یک فیلد سفارشی را برای یک پست خاص برمی‌گرداند.

برای ایجاد رابط‌های کاربری پیشرفته‌تر و کاربرپسندتر برای فیلدهای سفارشی (مانند انتخابگر تاریخ، آپلودر تصویر و...)، معمولاً از افزونه‌های قدرتمندی مانند Advanced Custom Fields (ACF) استفاده می‌شود.

۲۵. چگونه می‌توان یک منوی ناوبری (Navigation Menu) جدید در وردپرس ثبت و نمایش داد؟

این فرآیند نیز مشابه ثبت سایدبار است و دو مرحله دارد:

1. **ثبت مکان منو:** در فایل `functions.php`، با استفاده از تابع `register_nav_menus()`، مکان‌های منوی قالب خود را تعریف کنید. شما می‌توانید چندین مکان (مثلاً برای هدر، فوتر و...) ثبت کنید.

```
function register_my_menus() {
    register_nav_menus(
        array(
            'header-menu' => __( 'منوی اصلی' ),
            'footer-menu' => __( 'منوی فوتر' )
        )
    );
}
add_action( 'init', 'register_my_menus' );
```

2. **نمایش منو در قالب:** در فایل قالب مورد نظر (مثلاً header.php)، از تابع `wp_nav_menu()` برای نمایش منو استفاده کنید. شما باید نام مکانی که در مرحله قبل ثبت کردید را به عنوان پارامتر به این تابع بدهید.

```
wp_nav_menu( array( 'theme_location' => 'header-menu' ) );
```

پس از انجام این مراحل، کاربران می‌توانند از بخش «نمایش > فهرست‌ها» در پیشخوان، منوهای خود را بسازند و به این مکان‌ها اختصاص دهند.

۲۶. برای توسعه یک افزونه وردپرس، ساختار اصلی فایل افزونه و هدر اطلاعات آن باید چگونه باشد؟

برای ساخت یک افزونه ساده، کافی است یک فایل PHP در پوشه `wp-content/plugins` ایجاد کنید. مهم‌ترین بخش این فایل، **بلوک کامنت اطلاعات افزونه (Plugin Header)** در ابتدای آن است. وردپرس از این اطلاعات برای شناسایی و نمایش افزونه در پیشخوان استفاده می‌کند.

یک هدر استاندارد شامل موارد زیر است:

```

<?php
/**
 * Plugin Name: افزونه من
 * Plugin URI: https://example.com/plugins/the-basics/
 * Description: این اولین افزونه من است.
 * Version: 1.0.0
 * Author: نام شما
 * Author URI: https://author.example.com/
 * License: GPL v2 or later
 * License URI: https://www.gnu.org/licenses/gpl-2.0.html
 * Text Domain: my-plugin
 * Domain Path: /languages
 */

```

حداقل فیلد مورد نیاز `Plugin Name` است. پس از ایجاد این فایل، می‌توانید افزونه را در پیشخوان وردپرس فعال کرده و کدهای خود را به آن اضافه کنید.

۲۷. Nonce در وردپرس چیست و چرا برای امنیت فرم‌ها و URL‌ها اهمیت دارد؟

Nonce (Number used once) یک کلید امنیتی یکبار مصرف است که برای محافظت در برابر حملات **درخواست میان‌سایتی (CSRF - Cross-Site Request Forgery)** استفاده می‌شود. در این نوع حملات، یک هکر تلاش می‌کند تا یک کاربر لاگین‌شده را فریب دهد تا ناخواسته یک عملیات مخرب را در سایت شما انجام دهد (مثلاً حذف یک پست).

وردپرس با ایجاد یک Nonce منحصر به فرد برای هر کاربر و هر عملیات، از این حملات جلوگیری می‌کند. وقتی شما یک فرم یا یک URL حاوی عملیات حساس ایجاد می‌کنید، باید یک فیلد Nonce به آن اضافه کنید (با استفاده از تابع `wp_nonce_field()` یا `wp_nonce_url()`).

سپس، قبل از انجام آن عملیات، باید صحت Nonce را با استفاده از تابع `wp_verify_nonce()` بررسی کنید. اگر Nonce معتبر نباشد یا متعلق به کاربر دیگری باشد، وردپرس عملیات را متوقف می‌کند. این یک لایه امنیتی حیاتی برای هرگونه عملیات تغییر وضعیت در وردپرس است.

۲۸. چگونه می‌توان با استفاده از AJAX در وردپرس، اطلاعات را بدون بارگذاری مجدد صفحه، ارسال و دریافت کرد؟

وردپرس یک API داخلی برای مدیریت درخواست‌های AJAX به روشی امن و ساختاریافته فراهم می‌کند. مراحل اصلی به این صورت است:

1. **اسکرپت جاوااسکرپت:** شما باید یک فایل جاوااسکرپت بنویسید که درخواست AJAX را (معمولاً با استفاده از jQuery) به URL مخصوص AJAX در وردپرس (`wp-admin/admin-ajax.php`) ارسال کند. شما باید یک پارامتر `action` در درخواست خود مشخص کنید که نام هوک AJAX شما خواهد بود.

2. **محل‌سازی اسکرپت (Localization):** برای اینکه URL مربوط به `admin-ajax.php` و همچنین Nonce امنیتی را به صورت امن به اسکرپت خود منتقل کنید، باید از تابع `wp_localize_script()` در `functions.php` استفاده کنید.

3. **ایجاد توابع کنترل‌کننده (Handler Functions):** شما باید دو تابع PHP در `functions.php` یا فایل افزونه خود ایجاد کنید که به هوک‌های AJAX متصل شوند. یک هوک برای کاربران لاگین‌شده (`wp_ajax_{your_action}`) و یک هوک برای کاربران مهمان (`wp_ajax_nopriv_{your_action}`). این توابع درخواست را پردازش کرده و پاسخ را (معمولاً با فرمت JSON) برمی‌گردانند و در نهایت با `wp_die()` خاتمه می‌یابند.

این روش تضمین می‌کند که تمام درخواست‌های AJAX از طریق هسته وردپرس و با بررسی‌های امنیتی لازم پردازش شوند.

۲۹. تفاوت بین `get_template_part()` و `include()` در PHP چیست؟

هر دو برای وارد کردن یک فایل در فایل دیگر استفاده می‌شوند، اما `get_template_part()` روش مخصوص و بهینه وردپرس برای بارگذاری بخش‌های مختلف قالب است.

`include()` یک ساختار اصلی در PHP است که محتوای یک فایل را در فایل دیگر قرار می‌دهد. اگر فایل مورد نظر پیدا نشود، یک هشدار (Warning) تولید می‌کند و اجرای اسکرپت ادامه می‌یابد.

`get_template_part()` هوشمندتر عمل می‌کند. این تابع به شما اجازه می‌دهد تا یک «اسلاگ» (slug) و یک «نام» (name) را مشخص کنید. سپس وردپرس به دنبال فایل‌های منطبق در قالب فرزند (Child Theme) و سپس در قالب والد (Parent Theme) می‌گردد. برای مثال، `get_template_part('content', 'single')` ابتدا به دنبال `content-single.php` و اگر پیدا نشد، به دنبال `content.php` می‌گردد. این ویژگی، سفارشی‌سازی قالب‌ها را بسیار آسان‌تر و انعطاف‌پذیرتر می‌کند.

۳۰. چگونه می‌توان یک کلاس سفارشی به تگ `<body>` در صفحات مختلف اضافه کرد؟

وردپرس به صورت خودکار کلاس‌های بسیار مفیدی را به تگ `<body>` اضافه می‌کند (مانند `logged-in`، `admin-bar`، `single-post`، `page-id-123` و...). این کلاس‌ها برای استایل‌دهی اختصاصی صفحات مختلف بسیار کاربردی هستند.

برای افزودن کلاس سفارشی خود، می‌توانید از فیلتر هوک `body_class` استفاده کنید. شما یک تابع می‌نویسید که یک آرایه از کلاس‌های موجود را به عنوان ورودی می‌گیرد، کلاس سفارشی خود را به آن اضافه می‌کند و آرایه جدید را برمی‌گرداند.

```
add_filter( 'body_class', 'my_custom_body_class' );
function my_custom_body_class( $classes ) {
    if ( is_singular( 'portfolio' ) ) {
        $classes[] = 'portfolio-page';
        $classes[] = 'full-width-layout';
    }
    return $classes;
}
```

فراموش نکنید که در فایل `header.php`، تگ `<body>` باید به این صورت نوشته شده باشد: `<?php body_class`

۳.۱. Transient API در وردپرس چیست و چه زمانی باید از آن استفاده کرد؟

Transient API یک مکانیزم ساده برای کش کردن موقت اطلاعات در پایگاه داده است. این API به شما اجازه می‌دهد تا داده‌هایی که تولید آن‌ها زمان‌بر یا پرهزینه است (مانند نتایج یک کوئری پیچیده یا پاسخ یک API خارجی) را برای مدت زمان مشخصی ذخیره کنید.

این API دو تابع اصلی دارد:

- `set_transient('name', $value, $expiration)` : برای ذخیره کردن یک مقدار با یک نام مشخص و زمان انقضا (به ثانیه).
- `get_transient('name')` : برای بازیابی مقدار ذخیره شده. اگر مقدار وجود داشته باشد یا منقضی نشده باشد، برگردانده می‌شود. در غیر این صورت، `false` برمی‌گردد.

الگوی استفاده معمولاً به این صورت است: ابتدا با `get_transient()` بررسی می‌کنید که آیا داده در کش وجود دارد یا خیر. اگر وجود داشت، از آن استفاده می‌کنید. اگر وجود نداشت، داده را از منبع اصلی تولید می‌کنید، با `set_transient()` آن را در کش ذخیره می‌کنید و سپس از آن استفاده می‌کنید.

این روش به طور قابل توجهی عملکرد سایت را با کاهش درخواست‌های تکراری به پایگاه داده یا API‌های خارجی، بهبود می‌بخشد.

۳۲. چگونه می‌توان یک نقش کاربری سفارشی (Custom User Role) با دسترسی‌های مشخص ایجاد کرد؟

وردپرس به صورت پیش‌فرض نقش‌هایی مانند مدیر، ویرایشگر، نویسنده و... دارد. اما گاهی شما به یک نقش جدید با مجموعه‌ای از دسترسی‌های خاص نیاز دارید. برای مثال، یک نقش «مدیر فروشگاه» که فقط به بخش مدیریت محصولات ووکامرس دسترسی داشته باشد.

برای ایجاد نقش سفارشی، باید از تابع `add_role()` استفاده کنید. این تابع یک شناسه برای نقش، یک نام نمایشی و یک آرایه از دسترسی‌ها (Capabilities) را به عنوان ورودی می‌گیرد. شما باید این تابع را فقط یک بار اجرا کنید (معمولاً در زمان فعال‌سازی قالب یا افزونه) تا از ایجاد چندباره آن جلوگیری شود.

```
// این کد را در هوک فعال‌سازی افزونه قرار دهید
function add_my_custom_role() {
    add_role(
        'shop_manager',
        'مدیر فروشگاه',
        array(
            'read' => true, // دسترسی خواندن
            'edit_posts' => false, // عدم دسترسی به ویرایش پست‌ها
            'manage_woocommerce' => true, // دسترسی به مدیریت ووکامرس
            // سایر دسترسی‌ها ...
        )
    );
}
```

برای مدیریت پیشرفته‌تر نقش‌ها و دسترسی‌ها، می‌توان از افزونه‌هایی مانند User Role Editor استفاده کرد.

۳۳. تفاوت بین متا تگ‌های `noindex` و `nofollow` چیست و چگونه می‌توان آن‌ها را در وردپرس کنترل کرد؟

این دو متا تگ به موتورهای جستجو دستور می‌دهند که چگونه با یک صفحه یا یک لینک رفتار کنند.

- **noindex**: به ربات‌های موتور جستجو می‌گوید که این صفحه را در نتایج جستجوی خود ایندکس (فهرست) نکنند. این برای صفحاتی مانند صفحات تشکر، نتایج جستجوی داخلی سایت یا صفحات ادمین مفید است که نمی‌خواهید کاربران مستقیماً از طریق گوگل به آن‌ها برسند.
- **nofollow**: این تگ در سطح لینک (``) استفاده می‌شود و به موتورهای جستجو می‌گوید که اعتباری (Link Juice) را از صفحه فعلی به لینک مقصد منتقل نکنند.

نکنند. این برای لینک‌های تبلیغاتی، لینک‌های موجود در کامنت‌ها یا لینک به سایت‌هایی که به آن‌ها اعتماد کامل ندارید، استفاده می‌شود.

در وردپرس، می‌توانید از طریق تنظیمات «خواندن» < «نمایش به موتورهای جستجو» کل سایت را noindex کنید. برای کنترل دقیق‌تر در سطح هر صفحه، افزونه‌های سئو مانند Rank Math یا Yoast SEO ابزارهای کاملی را برای تنظیم این متا تگ‌ها ارائه می‌دهند.

۳۴. چگونه می‌توان یک تصویر را از طریق کد به کتابخانه رسانه وردپرس اضافه کرد؟

برای این کار باید از مجموعه‌ای از توابع وردپرس استفاده کنید. تابع اصلی `media_handle_upload()` یا `media_handle_sideload()` است. این توابع یک فایل را از یک منبع خارجی (`sideload`) یا از طریق فرم آپلود (`upload`) PHP دریافت کرده، آن را به پوشه آپلودهای وردپرس منتقل می‌کنند، یک رکورد در جدول `wp_posts` (با `post_type` برابر `attachment`) برای آن ایجاد می‌کنند و اطلاعات متای آن (مانند اندازه و نوع فایل) را تولید می‌کنند.

فرآیند کلی به این صورت است:

1. فایل را از منبع مورد نظر دانلود کرده و در یک مکان موقت ذخیره کنید.
2. فایل‌های مورد نیاز وردپرس را با `require_once(ABSPATH . 'wp-admin/includes/image.php')` و ... فراخوانی کنید.
3. یک آرایه حاوی اطلاعات فایل (نام، مسیر موقت) بسازید.
4. تابع `media_handle_sideload()` را با این آرایه و شناسه پستی که می‌خواهید این رسانه به آن پیوست شود، فراخوانی کنید.
5. این تابع در صورت موفقیت، شناسه (ID) پیوست جدید را برمی‌گرداند.

۳۵. هدف از وجود فایل `readme.txt` در یک افزونه وردپرس چیست؟

فایل `readme.txt` که با یک سینتکس استاندارد Markdown نوشته می‌شود، برای نمایش اطلاعات افزونه شما در مخزن رسمی افزونه‌های وردپرس ([WordPress.org](https://wordpress.org)) استفاده می‌شود. این فایل منبع اصلی اطلاعاتی است که کاربران در صفحه افزونه شما مشاهده می‌کنند.

این فایل شامل بخش‌های مختلفی است:

- **هدر اطلاعات:** نام افزونه، مشارکت‌کنندگان، نسخه مورد نیاز وردپرس و ...
- **Description:** توضیحات کامل درباره کاری که افزونه انجام می‌دهد.
- **Installation:** راهنمای قدم به قدم نصب.

- **Frequently Asked Questions (FAQ):** سوالات متداول.
- **Screenshots:** لیست تصاویری که در صفحه افزونه نمایش داده می‌شوند.
- **Changelog:** تاریخچه تغییرات در هر نسخه.

داشتن یک فایل `readme.txt` کامل و دقیق برای موفقیت یک افزونه در مخزن وردپرس ضروری است.

بخش سوم: امنیت، عملکرد و عیب‌یابی

۳۶. تفاوت بین احراز هویت (Authentication) و مجوز دسترسی (Authorization) در وردپرس چیست؟

این دو مفهوم امنیتی اغلب با هم اشتباه گرفته می‌شوند.

- **احراز هویت (Authentication):** فرآیند تأیید هویت یک کاربر است. یعنی بررسی اینکه «شما چه کسی هستید؟». در وردپرس، این کار معمولاً از طریق نام کاربری و رمز عبور انجام می‌شود. اگر اطلاعات صحیح باشد، وردپرس هویت شما را تأیید می‌کند.
- **مجوز دسترسی (Authorization):** فرآیندی است که پس از احراز هویت موفق انجام می‌شود. این فرآیند مشخص می‌کند که «شما اجازه انجام چه کارهایی را دارید؟». وردپرس این کار را از طریق سیستم نقش‌ها و دسترسی‌ها (Roles and Capabilities) مدیریت می‌کند. برای مثال، یک کاربر با نقش «نویسنده» احراز هویت می‌شود، اما فقط مجاز به ایجاد و ویرایش پست‌های خودش است و نمی‌تواند افزونه‌ها را مدیریت کند.

۳۷. حمله Brute Force چیست و چگونه می‌توان در وردپرس از آن جلوگیری کرد؟

حمله Brute Force (حمله جستجوی فراگیر) یک روش هک است که در آن، یک ربات یا اسکریپت به صورت خودکار هزاران یا میلیون‌ها ترکیب مختلف از نام کاربری و رمز عبور را امتحان می‌کند تا بتواند به پیشخوان سایت شما وارد شود. این یکی از شایع‌ترین حملات علیه سایت‌های وردپرس است.

روش‌های جلوگیری:

1. **محدود کردن تلاش‌های ورود:** این مهم‌ترین روش است. افزونه‌هایی مانند Limit Login Attempts Reloaded یا افزونه‌های امنیتی جامع، پس از چند تلاش ناموفق، IP مهاجم را برای مدتی مسدود می‌کنند.
2. **استفاده از رمزهای عبور قوی:** رمزهای پیچیده، شکستن آن‌ها را برای مهاجم بسیار دشوارتر می‌کند.

3. **تغییر URL صفحه ورود:** با تغییر آدرس `wp-login.php/` به یک آدرس منحصر به فرد، ربات‌ها را در پیدا کردن صفحه ورود ناکام می‌گذارید.
4. **استفاده از احراز هویت دو مرحله‌ای (2FA):** حتی اگر مهاجم رمز عبور شما را پیدا کند، برای ورود به کد یکبار مصرفی که به موبایل شما ارسال می‌شود، نیاز خواهد داشت.
5. **استفاده از CAPTCHA:** افزودن یک چالش کپچا به فرم ورود، ربات‌ها را متوقف می‌کند.

۳۸. چگونه می‌توان از پایگاه داده وردپرس نسخه پشتیبان (Backup) تهیه کرد؟

پایگاه داده قلب سایت وردپرسی شماست و شامل تمام محتوا، تنظیمات و اطلاعات کاربران است. تهیه نسخه پشتیبان منظم از آن حیاتی است.

- **استفاده از افزونه‌های پشتیبان‌گیری:** این ساده‌ترین و بهترین روش است. افزونه‌هایی مانند UpdraftPlus، Duplicator یا All-in-One WP Migration به شما اجازه می‌دهند تا به صورت خودکار و زمان‌بندی شده از کل سایت (فایل‌ها و پایگاه داده) نسخه پشتیبان تهیه کرده و آن را در یک فضای ذخیره‌سازی ابری (مانند Google Drive یا Dropbox) ذخیره کنید.
- **استفاده از phpMyAdmin:** شما می‌توانید از طریق کنترل پنل هاست خود وارد phpMyAdmin شوید، پایگاه داده وردپرس خود را انتخاب کرده و از تب Export، یک خروجی با فرمت SQL از تمام جداول آن بگیرید.
- **استفاده از دستورات WP-CLI:** برای کاربران حرفه‌ای، WP-CLI دستور `wp db export` را برای گرفتن خروجی سریع از پایگاه داده فراهم می‌کند.

۳۹. خطای «صفحه سفید مرگ» (White Screen of Death - WSoD) چیست و مراحل عیب‌یابی آن کدامند؟

صفحه سفید مرگ یکی از معروف‌ترین خطاهای وردپرس است که در آن، سایت شما یک صفحه کاملاً سفید و خالی را بدون هیچ پیام خطایی نمایش می‌دهد. این خطا معمولاً به دلیل مشکلات PHP یا خطاهای پایگاه داده رخ می‌دهد.

مراحل عیب‌یابی:

1. **فعال کردن حالت دیباگ:** اولین قدم، فعال کردن `WP_DEBUG` در فایل `wp-config.php` است تا پیام خطای واقعی نمایش داده شود. این پیام معمولاً به شما می‌گوید که مشکل از کدام فایل و کدام خط کد است.
2. **غیرفعال کردن تمام افزونه‌ها:** یکی از شایع‌ترین دلایل این خطا، تداخل بین افزونه‌هاست. با تغییر نام پوشه `plugins` در `wp-content` (مثلاً به `plugins_old`) از طریق FTP، می‌توانید تمام افزونه‌ها را غیرفعال کنید. اگر سایت بالا آمد، مشکل از یکی از افزونه‌ها بوده است. سپس می‌توانید افزونه‌ها را یکی‌یکی فعال کنید تا افزونه مشکل‌ساز را پیدا کنید.

3. **تغییر به قالب پیش فرض وردپرس:** ممکن است مشکل از قالب شما باشد. با تغییر نام پوشه قالب فعلی خود، وردپرس را مجبور کنید تا از یک قالب پیش فرض (مانند Twenty Twenty-Four) استفاده کند. اگر سایت درست شد، مشکل در کدهای قالب شماست.

4. **افزایش محدودیت حافظه PHP:** گاهی اوقات این خطا به دلیل کمبود حافظه برای اجرای یک اسکریپت رخ می‌دهد. می‌توانید با افزودن `define('WP_MEMORY_LIMIT', '256M');` به `wp-config.php` این محدودیت را افزایش دهید.

۴۰. CDN (Content Delivery Network) چیست و چگونه به بهبود عملکرد یک سایت وردپرسی کمک می‌کند؟

CDN شبکه‌ای از سرورهاست که در نقاط مختلف جغرافیایی جهان توزیع شده‌اند. هدف اصلی CDN، افزایش سرعت تحویل محتوای سایت به کاربران است.

وقتی شما از CDN استفاده می‌کنید، یک کپی از فایل‌های ثابت (Static Files) سایت شما مانند تصاویر، فایل‌های CSS و JavaScript در تمام این سرورها ذخیره (کش) می‌شود. هنگامی که یک کاربر از سایت شما بازدید می‌کند، این فایل‌ها از نزدیک‌ترین سرور CDN به موقعیت جغرافیایی او برایش ارسال می‌شوند، نه از سرور اصلی هاست شما.

این کار باعث می‌شود:

- **زمان بارگذاری صفحه (Page Load Time) به شدت کاهش یابد:** زیرا فاصله فیزیکی بین کاربر و سرور کمتر می‌شود.
- **بار روی سرور اصلی شما کاهش یابد:** زیرا بخش زیادی از درخواست‌ها توسط سرورهای CDN پاسخ داده می‌شوند.
- **امنیت و پایداری سایت افزایش یابد:** بسیاری از CDN‌ها خدمات محافظت در برابر حملات DDoS را نیز ارائه می‌دهند.

سرویس‌های معروفی مانند Cloudflare یا BunnyCDN با وردپرس یکپارچه می‌شوند.

۴۱. کش کردن (Caching) در وردپرس به چه معناست و انواع مختلف آن کدامند؟

کش کردن فرآیند ذخیره‌سازی موقت یک نسخه از خروجی سایت شماست تا در درخواست‌های بعدی، به جای پردازش مجدد و سنگین PHP و کوئری‌های پایگاه داده، همان نسخه آماده به سرعت به کاربر تحویل داده شود. این یکی از مؤثرترین راه‌ها برای افزایش سرعت سایت است.

انواع اصلی کش:

- **کش صفحه (Page Caching):** کل خروجی HTML یک صفحه ذخیره می‌شود. این کار توسط افزونه‌های کش مانند WP Rocket یا W3 Total Cache انجام می‌شود.

- **کش مرورگر (Browser Caching):** به مرورگر کاربر دستور می‌دهد تا فایل‌های ثابت (تصاویر، CSS، JS) را برای مدتی در کامپیوتر خود ذخیره کند تا در بازدیدهای بعدی نیازی به دانلود مجدد آن‌ها نباشد.
- **کش پایگاه داده/شیء (Database/Object Caching):** نتایج کوئری‌های متداول پایگاه داده به صورت موقت در حافظه (RAM) سرور ذخیره می‌شوند تا از اجرای چندباره آن‌ها جلوگیری شود. این کار با استفاده از ابزارهایی مانند Redis یا Memcached و افزونه‌های مربوطه انجام می‌شود.

۴۲. چگونه می‌توان تصاویر را برای وب بهینه‌سازی کرد تا سرعت سایت افزایش یابد؟

تصاویر معمولاً سنگین‌ترین بخش یک صفحه وب هستند و بهینه‌سازی آن‌ها تأثیر چشمگیری بر سرعت سایت دارد.

1. **انتخاب فرمت مناسب:** از JPEG برای عکس‌ها و تصاویر با رنگ‌های زیاد، از PNG برای تصاویر با پس‌زمینه شفاف (Transparent) و از WebP به عنوان یک فرمت مدرن و کم‌حجم برای هر دو حالت استفاده کنید. برای انیمیشن‌های ساده، GIF یا ویدئوهای کوتاه بهتر است.
2. **تغییر اندازه صحیح (Resizing):** قبل از آپلود، اندازه تصویر را متناسب با حداکثر اندازه‌ای که در سایت نمایش داده می‌شود، تغییر دهید. آپلود یک عکس 5000 پیکسلی برای نمایش در یک کادر 800 پیکسلی، اتلاف منابع است.
3. **فشرده‌سازی (Compression):** با استفاده از ابزارهای آنلاین (مانند TinyPNG) یا افزونه‌های وردپرس (مانند Smush یا ShortPixel)، حجم فایل تصویر را بدون افت کیفیت محسوس، کاهش دهید.
4. **پیاده‌سازی بارگذاری تنبل (Lazy Loading):** با این تکنیک، تصاویری که در دید کاربر نیستند (مثلاً در پایین صفحه)، فقط زمانی بارگذاری می‌شوند که کاربر به نزدیکی آن‌ها اسکرول کند. وردپرس از نسخه 5.5 به بعد این ویژگی را به صورت خودکار برای تصاویر اعمال می‌کند.

۴۳. خطای Internal Server Error 500 به چه معناست و چگونه می‌توان آن را رفع کرد؟

این یک خطای عمومی سمت سرور است که نشان می‌دهد مشکلی در سرور وب‌سایت شما رخ داده، اما سرور نمی‌تواند دقیقاً بگوید که مشکل چیست. در وردپرس، این خطا اغلب به دلایل زیر رخ می‌دهد:

- **مشکل در فایل htaccess:** یک کد اشتباه یا خراب در این فایل می‌تواند باعث این خطا شود. با تغییر نام فایل htaccess و سپس ذخیره مجدد تنظیمات پیوندهای یکتا در پیشخوان، می‌توانید یک فایل سالم جدید ایجاد کنید.
- **تداخل افزونه یا قالب:** مشابه خطای صفحه سفید، یک افزونه یا قالب مشکل‌دار می‌تواند این خطا را ایجاد کند. مراحل عیب‌یابی مشابه (غیرفعال کردن افزونه‌ها و تغییر قالب) است.

- **رسیدن به محدودیت حافظه PHP:** اگر یک اسکریپت به حافظه بیشتری از آنچه در دسترس است نیاز داشته باشد، این خطا ممکن است رخ دهد. افزایش `WP_MEMORY_LIMIT` می‌تواند مشکل را حل کند.
- **نسخه PHP ناسازگار:** ممکن است یک افزونه یا قالب به نسخه جدیدتری از PHP نیاز داشته باشد که روی سرور شما نصب نیست.
- بررسی لاگ خطاهای سرور (Error Logs) معمولاً اطلاعات دقیق‌تری درباره منشأ این خطا ارائه می‌دهد.

۴۴. WP-CLI چیست و چند نمونه از دستورات کاربردی آن را نام ببرید.

WP-CLI (WordPress Command-Line Interface) یک ابزار خط فرمان برای مدیریت سایت‌های وردپرسی است. با استفاده از آن، می‌توانید بدون نیاز به مرورگر وب و پیشخوان گرافیکی، بسیاری از وظایف مدیریتی را از طریق ترمینال انجام دهید. این ابزار برای توسعه‌دهندگان و مدیران سرور بسیار کارآمد است.

چند دستور کاربردی:

- `wp core update`: هسته وردپرس را به‌روزرسانی می‌کند.
- `wp plugin install woocommerce --activate`: افزونه ووکامرس را نصب و فعال می‌کند.
- `wp theme list`: لیست تمام قالب‌های نصب‌شده را نمایش می‌دهد.
- `wp user create bob bob@example.com --role=editor`: یک کاربر جدید با نقش ویرایشگر ایجاد می‌کند.
- `wp db export backup.sql`: از پایگاه داده یک خروجی با نام `backup.sql` می‌گیرد.
- `wp search-replace 'http://example.test' 'https://example.com'`: یک رشته را در کل پایگاه داده جستجو و جایگزین می‌کند (بسیار مفید هنگام انتقال سایت).
- `wp post generate --count=10`: ده پست آزمایشی ایجاد می‌کند.

۴۵. چگونه می‌توان وردپرس را مجبور به استفاده از HTTPS کرد؟

پس از نصب گواهی SSL روی هاست خود، باید وردپرس را پیکربندی کنید تا تمام ترافیک را از طریق HTTPS (اتصال امن) هدایت کند.

1. **تغییر آدرس‌های وردپرس:** در پیشخوان، به بخش «تنظیمات > عمومی» بروید و آدرس وردپرس (URL) و آدرس سایت (URL) را از `http://` به `https://` تغییر دهید.
2. **ریدایرکت در htaccess:** برای اینکه کاربرانی که با آدرس `http` وارد می‌شوند به صورت خودکار به `https` هدایت شوند، کد زیر را به فایل `htaccess` خود اضافه کنید:

```
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
</IfModule>
```

3. **رفع خطای محتوای ترکیبی (Mixed Content):** پس از فعال‌سازی HTTPS، ممکن است برخی منابع (مانند تصاویر یا اسکریپت‌ها) هنوز با آدرس http بارگذاری شوند که باعث نمایش اخطار امنیتی در مرورگر می‌شود. باید این آدرس‌ها را در پایگاه داده (با افزونه‌هایی مانند Better Search Replace) یا در کدهای قالب اصلاح کنید.

۴۶. نقش Cron Jobs در وردپرس (WP-Cron) چیست و چه محدودیت‌هایی دارد؟

WP-Cron سیستمی است که وردپرس برای اجرای وظایف زمان‌بندی‌شده (Scheduled Tasks) از آن استفاده می‌کند. کارهایی مانند بررسی برای به‌روزرسانی‌ها، انتشار پست‌های زمان‌بندی‌شده یا اجرای بک‌آپ‌های خودکار از طریق WP-Cron انجام می‌شود.

اما WP-Cron یک Cron Job واقعی در سطح سرور نیست. این سیستم وابسته به بازدید کاربران است. یعنی هر بار که یک صفحه از سایت شما بارگذاری می‌شود، WP-Cron بررسی می‌کند که آیا وظیفه زمان‌بندی‌شده‌ای برای اجرا وجود دارد یا خیر.

محدودیت اصلی این است که اگر سایت شما بازدیدکننده‌ای نداشته باشد، هیچ وظیفه‌ای اجرا نخواهد شد. برای سایت‌های کم‌بازدید، این می‌تواند باعث تأخیر در اجرای وظایف مهم شود. برای سایت‌های پر بازدید نیز، اجرای مکرر WP-Cron در هر بار بازدید می‌تواند بار اضافی روی سرور ایجاد کند.

راه حل بهتر، غیرفعال کردن WP-Cron (با افزودن `define('DISABLE_WP_CRON', true);` به `wp-config.php`) و ایجاد یک Cron Job واقعی در کنترل پنل هاست است که به صورت مستقیم و در فواصل زمانی معین، فایل `wp-cron.php` را فراخوانی کند.

۴۷. چگونه می‌توان یک سایت وردپرسی را به یک دامنه یا هاست جدید منتقل کرد (Migration)?

انتقال سایت یک فرآیند حساس است. روش کلی به این صورت است:

1. **پشتیبان‌گیری کامل:** از تمام فایل‌های سایت (از طریق FTP) و کل پایگاه داده (از طریق phpMyAdmin) یک نسخه پشتیبان کامل تهیه کنید.

2. **انتقال فایل‌ها:** تمام فایل‌های وردپرس را در هاست جدید آپلود کنید.

3. **ایجاد پایگاه داده جدید:** در هاست جدید، یک پایگاه داده جدید به همراه یک کاربر و رمز عبور برای آن ایجاد کنید.

4. **وارد کردن پایگاه داده:** فایل SQL که در مرحله ۱ خروجی گرفته بودید را در پایگاه داده جدید وارد (Import) کنید.

5. **به روزرسانی wp-config.php:** فایل wp-config.php را در هاست جدید ویرایش کرده و اطلاعات اتصال به پایگاه داده جدید (نام، کاربر و رمز عبور) را در آن وارد کنید.

6. **به روزرسانی آدرسها در پایگاه داده:** آدرس دامنه قدیمی باید در سراسر پایگاه داده با آدرس دامنه جدید جایگزین شود. این کار را می‌توان با اسکریپت‌هایی مانند Search Replace DB یا با دستور wp search-replace در WP-CLI انجام داد. تغییر دستی این مقادیر توصیه نمی‌شود زیرا ممکن است داده‌های سریالی (Serialized Data) را خراب کند.

روش ساده‌تر، استفاده از **افزونه‌های انتقال** مانند Duplicator یا All-in-One WP Migration است. این افزونه‌ها تمام مراحل بالا را به صورت خودکار در قالب یک پکیج انجام می‌دهند و فرآیند انتقال را بسیار آسان‌تر می‌کنند.

۴۸. وردپرس چند سایت (Multisite) چیست و در چه سناریوهایی کاربرد دارد؟

وردپرس چند سایت یک ویژگی پیشرفته در وردپرس است که به شما اجازه می‌دهد تا یک شبکه از چندین سایت را از طریق یک نصب واحد وردپرس مدیریت کنید. تمام سایت‌های این شبکه از یک هسته وردپرس، قالب‌ها و افزونه‌های مشترک استفاده می‌کنند، اما هر کدام پوشه آپلود و جداول پایگاه داده مجزای خود را برای محتوا و کاربران دارند.

کاربردها:

- **دانشگاه‌ها یا شرکت‌های بزرگ:** که می‌خواهند برای هر دپارتمان یا شعبه، یک سایت جداگانه اما تحت مدیریت متمرکز داشته باشند (مثلاً science.university.edu و art.university.edu).

- **سرویس‌های وبلاگ‌دهی:** مانند wordpress.com که به هر کاربر اجازه می‌دهد سایت خود را بسازد (user1.wordpress.com).

- **سایت‌های چندزبانه:** که برای هر زبان یک سایت جداگانه در شبکه ایجاد می‌کنند.

مدیر اصلی شبکه (Super Admin) می‌تواند قالب‌ها و افزونه‌ها را برای کل شبکه نصب کند و مدیران هر سایت فقط می‌توانند آن‌ها را برای سایت خود فعال یا غیرفعال کنند.

۴۹. چگونه می‌توان از بارگذاری فونت‌های گوگل (Google Fonts) توسط قالب یا افزونه‌ها جلوگیری کرد؟

بسیاری از قالب‌ها و افزونه‌ها برای زیبایی بیشتر از فونت‌های گوگل استفاده می‌کنند. اما بارگذاری این فونت‌ها از سرورهای گوگل می‌تواند باعث کندی سایت (به‌ویژه برای کاربران ایرانی به دلیل تحریم‌ها) و نگرانی‌های مربوط به حریم خصوصی (GDPR) شود. برای غیرفعال کردن آن‌ها:

- **گزینه‌های قالب/افزونه:** بسیاری از قالب‌ها و افزونه‌های مدرن، گزینه‌ای برای غیرفعال کردن فونت‌های گوگل در تنظیمات خود دارند. این اولین و ساده‌ترین راه است.
 - **استفاده از افزونه:** افزونه‌هایی مانند `Disable and Remove Google Fonts` یا `Asset CleanUp` می‌توانند به صورت خودکار درخواست‌های مربوط به فونت‌های گوگل را شناسایی و مسدود کنند.
 - **کدنویسی:** با پیدا کردن «هندل» (`handle`) استایل مربوط به فونت گوگل، می‌توانید آن را با استفاده از تابع `wp_dequeue_style()` در فایل `functions.php` از صف بارگذاری خارج کنید.
- پس از غیرفعال کردن، باید فونت‌های جایگزین را به صورت محلی (`Self-host`) در سایت خود بارگذاری کرده و با استفاده از `CSS`، آن‌ها را اعمال کنید.

۵۰. تفاوت بین کوکی (Cookie) و سشن (Session) در بستر وب چیست؟

هر دو برای ذخیره‌سازی اطلاعات کاربر در طول بازدید از یک وب‌سایت استفاده می‌شوند، اما محل ذخیره‌سازی و چرخه عمر آن‌ها متفاوت است.

- **کوکی (Cookie):** یک فایل متنی کوچک است که توسط وب‌سایت **در مرورگر کاربر** ذخیره می‌شود. کوکی‌ها می‌توانند برای مدت طولانی (حتی پس از بستن مرورگر) باقی بمانند و برای کارهایی مانند به خاطر سپردن اطلاعات ورود (`remember me`)، ردیابی رفتار کاربر و شخصی‌سازی محتوا استفاده می‌شوند. وردپرس از کوکی‌ها برای مدیریت سشن‌های ورود کاربران استفاده می‌کند.
- **سشن (Session):** اطلاعات کاربر **در سرور** ذخیره می‌شود. یک شناسه منحصر به فرد (`Session ID`) از طریق یک کوکی به مرورگر کاربر ارسال می‌شود و در درخواست‌های بعدی، سرور از طریق این شناسه، اطلاعات مربوط به آن کاربر را بازیابی می‌کند. سشن‌ها معمولاً با بستن مرورگر از بین می‌روند و برای ذخیره‌سازی اطلاعات حساس‌تر (مانند محتوای سبد خرید) مناسب‌تر هستند زیرا داده‌ها در سمت کاربر ذخیره نمی‌شوند.

۵۱. چگونه می‌توان یک قالب یا افزونه را ترجمه و بومی‌سازی کرد؟

وردپرس از سیستم `gettext` برای بین‌المللی‌سازی (`Internationalization - i18n`) و محلی‌سازی (`Localization - l10n`) استفاده می‌کند. برای ترجمه یک قالب یا افزونه که به درستی آماده‌سازی شده باشد:

1. **فایل POT (Portable Object Template)**: توسعه‌دهنده باید تمام رشته‌های متنی قابل ترجمه را با توابع خاصی مانند `__()` یا `e_()` مشخص کرده باشد. سپس یک فایل `pot` تولید می‌شود که حاوی لیست تمام این رشته‌ها به زبان اصلی (انگلیسی) است.
2. **ایجاد فایل PO (Portable Object)**: شما باید این فایل `pot` را با استفاده از یک نرم‌افزار ترجمه مانند Poedit باز کنید. سپس آن را برای زبان مقصد خود (مثلاً فارسی - `fa_IR`) ذخیره کنید. این کار یک فایل `po` ایجاد می‌کند.
3. **ترجمه رشته‌ها**: در نرم‌افزار Poedit، شما تمام رشته‌های انگلیسی را مشاهده کرده و معادل فارسی آن‌ها را وارد می‌کنید.
4. **ایجاد فایل MO (Machine Object)**: پس از ذخیره فایل `po`، نرم‌افزار به صورت خودکار یک فایل `mo` کامپایل می‌کند. این فایل، نسخه باینری و بهینه‌شده ترجمه‌هاست که وردپرس برای استفاده در سایت، آن را می‌خواند.
5. **قرار دادن فایل‌ها در مسیر صحیح**: شما باید فایل‌های `fa_IR.mo` و `fa_IR.po` را در پوشه `languages` یا قالب `lang` یا افزونه مورد نظر قرار دهید. اگر وردپرس شما روی زبان فارسی تنظیم شده باشد، ترجمه‌ها به صورت خودکار اعمال خواهند شد.

۵۲. Heartbeat API در وردپرس چیست و چگونه می‌توان آن را مدیریت کرد؟

Heartbeat API یک ویژگی در وردپرس است که به مرورگر اجازه می‌دهد تا به صورت منظم (هر 15 تا 60 ثانیه) با سرور ارتباط برقرار کرده و داده‌ها را همگام‌سازی کند. این API برای کارهایی مانند ذخیره خودکار پیش‌نویس‌ها، نمایش قفل بودن یک پست (وقتی کاربر دیگری در حال ویرایش آن است) و نمایش اعلان‌های لحظه‌ای در پیشخوان استفاده می‌شود.

مشکل اینجاست که این درخواست‌های مداوم AJAX (به `admin-ajax.php`) می‌توانند بار زیادی روی سرور ایجاد کنند، به‌ویژه در هاست‌های اشتراکی یا زمانی که چندین تب پیشخوان به صورت همزمان باز هستند.

برای مدیریت آن:

- **استفاده از افزونه**: افزونه‌هایی مانند Heartbeat Control به شما اجازه می‌دهند تا فرکانس اجرای Heartbeat را کاهش دهید (مثلاً هر 300 ثانیه) یا آن را در بخش‌های خاصی از سایت (مانند فرانت‌اند) به طور کامل غیرفعال کنید. این بهترین و ساده‌ترین روش است.
- **کدنویسی**: می‌توان با استفاده از هوک‌های مربوطه در `functions.php` نیز رفتار Heartbeat API را کنترل کرد.

۵۳. چگونه می‌توان از اجرای فایل‌های PHP در پوشه آپلودهای وردپرس (/wp-content/uploads) جلوگیری کرد؟

پوشه آپلودها برای ذخیره فایل‌های رسانه‌ای است. به دلایل امنیتی، هیچ فایل PHP نباید در این پوشه اجرا شود. اگر یک هکر بتواند یک فایل PHP مخرب (شل) را در این پوشه آپلود کند، می‌تواند کنترل کامل سایت شما را در دست بگیرد.

برای جلوگیری از این امر، می‌توانید یک فایل `htaccess` در داخل پوشه `/wp-content/uploads/` ایجاد کرده و کد زیر را در آن قرار دهید:

```
<Files *.php>
deny from all
</Files>
```

این کد به وب‌سرور آچای دستور می‌دهد که دسترسی مستقیم به تمام فایل‌هایی که با پسوند `.php` خاتمه می‌یابند را مسدود کند. این یک اقدام امنیتی ساده و بسیار مؤثر است.

۵۴. Object Caching چیست و چه زمانی استفاده از آن ضروری می‌شود؟

Object Caching یک روش کش سمت سرور است که نتایج کوئری‌های پیچیده و زمان‌بر پایگاه داده را در حافظه RAM سرور (که بسیار سریع‌تر از دیسک است) ذخیره می‌کند. وردپرس به صورت پیش‌فرض یک سیستم Object Cache داخلی دارد، اما این سیستم غیرمداوم (non-persistent) است، یعنی کش فقط در طول یک درخواست صفحه باقی می‌ماند و در درخواست بعدی از بین می‌رود.

برای سایت‌های بزرگ، پرترافیک و پیچیده (مانند فروشگاه‌های ووکامرس یا شبکه‌های اجتماعی)، استفاده از یک سیستم Object Cache مداوم (persistent) مانند `Redis` یا `Memcached` ضروری است. این سیستم‌ها به عنوان یک سرویس جداگانه روی سرور اجرا می‌شوند و به وردپرس اجازه می‌دهند تا داده‌های کش شده را بین درخواست‌های مختلف به اشتراک بگذارد.

این کار به طور چشمگیری تعداد کوئری‌های ارسالی به پایگاه داده MySQL را کاهش داده و باعث افزایش فوق‌العاده سرعت و مقیاس‌پذیری سایت می‌شود.

۵۵. چگونه می‌توان تمام ایمیل‌های ارسالی از وردپرس را لاگ یا مشاهده کرد؟

وردپرس از تابع `wp_mail()` برای ارسال تمام ایمیل‌ها (مانند اعلان ثبت‌نام کاربر جدید، بازنشانی رمز عبور، اعلان‌های ووکامرس و...) استفاده می‌کند. گاهی اوقات برای عیب‌یابی مشکلات ارسال ایمیل، نیاز دارید که ببینید وردپرس دقیقاً چه ایمیل‌هایی را با چه محتوایی ارسال می‌کند.

بهترین راه برای این کار، استفاده از یک افزونه لاگ ایمیل مانند **WP Mail Logging** است. این افزونه بدون ایجاد اختلال در فرآیند ارسال، یک کپی از تمام ایمیل‌های ارسالی را در پیشخوان وردپرس ذخیره می‌کند. شما می‌توانید محتوای کامل، هدرها، گیرندگان و هرگونه خطای احتمالی در ارسال را مشاهده کنید.

روش دیگر، استفاده از یک سرویس ارسال ایمیل تراکنشی (Transactional Email Service) مانند SendGrid یا Mailgun است. این سرویس‌ها علاوه بر افزایش قابلیت اطمینان در تحویل ایمیل‌ها، داشبوردهای کاملی برای ردیابی و مشاهده تمام ایمیل‌های ارسالی ارائه می‌دهند.

۵۶. تفاوت بین `get_option()`، `get_theme_mod` و `get_post_meta()`

چیست؟

این سه تابع برای بازیابی مقادیر ذخیره‌شده در پایگاه داده استفاده می‌شوند، اما از مکان‌های مختلفی داده‌ها را می‌خوانند:

- **`get_option()`**: برای خواندن مقادیر از جدول `wp_options` استفاده می‌شود. این جدول برای ذخیره تنظیمات کلی سایت، تنظیمات افزونه‌ها و داده‌هایی که به پست یا قالب خاصی مرتبط نیستند، به کار می‌رود.
- **`get_theme_mod()`**: یک تابع پوششی برای `get_option()` است که مقادیر را به صورت اختصاصی برای **قالب فعال فعلی** ذخیره و بازیابی می‌کند. این تابع برای ذخیره تنظیماتی که از طریق «سفارشی‌ساز قالب» (Customizer) تنظیم می‌شوند، استفاده می‌شود. اگر کاربر قالب را عوض کند، این تنظیمات دیگر اعمال نخواهند شد.
- **`get_post_meta()`**: برای خواندن مقادیر از جدول `wp_postmeta` استفاده می‌شود. این جدول برای ذخیره اطلاعات اضافی و فراداده‌های مربوط به یک پست، برگه یا پست تایپ سفارشی خاص به کار می‌رود (همان فیلدهای سفارشی).

۵۷. چگونه می‌توان یک صفحه با قالب سفارشی (Custom Page Template) ایجاد کرد؟

گاهی اوقات شما نیاز دارید که یک یا چند برگه در سایت شما، ظاهری کاملاً متفاوت از سایر برگه‌ها داشته باشند. برای این کار می‌توانید یک قالب صفحه سفارشی ایجاد کنید:

1. یک فایل PHP جدید در پوشه قالب خود ایجاد کنید (مثلاً `template-full-width.php`).

2. در ابتدای این فایل، یک بلوک کامنت با نام `Template Name` قرار دهید. این نام در ویرایشگر برگه نمایش داده خواهد شد.

```
*/ php <?php`
```

○ Template Name: قالب تمام-عرض / * <? /

3. کدهای HTML و PHP مورد نظر خود را برای طراحی این قالب سفارشی در این فایل بنویسید. شما می‌توانید حلقه وردپرس را برای نمایش عنوان و محتوای برگه در آن قرار دهید.

4. حالا به پیشخوان وردپرس بروید و یک برگه جدید ایجاد کنید یا یک برگه موجود را ویرایش کنید. در بخش «ویژگی‌های برگه» (Page Attributes)، یک منوی کشویی به نام «قالب» (Template) ظاهر می‌شود. شما می‌توانید نام قالبی که در مرحله ۲ تعریف کردید را از این لیست انتخاب کنید. با این کار، آن برگه با استفاده از فایل قالب سفارشی شما نمایش داده خواهد شد.

۵۸. چگونه می‌توان یک سایت وردپرسی را در حالت تعمیر و نگهداری (Maintenance Mode) قرار داد؟

وقتی در حال انجام تغییرات اساسی روی سایت خود هستید، بهتر است آن را در حالت تعمیر قرار دهید تا کاربران با یک سایت ناقص یا خراب مواجه نشوند.

- **روش داخلی وردپرس:** وقتی شما هسته، قالب یا افزونه‌ها را از پیشخوان به‌روزرسانی می‌کنید، وردپرس به صورت خودکار یک فایل به نام `maintenance` در ریشه سایت ایجاد می‌کند. تا زمانی که این فایل وجود داشته باشد، سایت یک پیام پیش‌فرض «به‌زودی برمی‌گردیم» را نمایش می‌دهد. پس از اتمام به‌روزرسانی، این فایل به صورت خودکار حذف می‌شود. گاهی اوقات در صورت شکست آپدیت، این فایل باقی می‌ماند و باید آن را به صورت دستی حذف کنید.
- **استفاده از افزونه:** این روش بهتری است زیرا به شما کنترل بیشتری می‌دهد. افزونه‌هایی مانند **WP Maintenance Mode** یا **SeedProd** به شما اجازه می‌دهند تا صفحات «به‌زودی» یا «در دست تعمیر» بسیار زیبا و سفارشی با فرم جمع‌آوری ایمیل، شمارش معکوس و لینک‌های شبکه‌های اجتماعی طراحی کنید. همچنین به شما (به عنوان مدیر لاگین شده) اجازه می‌دهند تا سایت اصلی را مشاهده کنید.
- **کدنویسی در `functions.php`:** می‌توان با استفاده از هوک `template_redirect` و بررسی نقش کاربر، کاربران عادی را به یک صفحه تعمیر هدایت کرد.

۵۹. Salt Keys در فایل `wp-config.php` چه هستند و چه اهمیتی دارند؟

Salt Keys (کلیدها و نمک‌های امنیتی) مجموعه‌ای از هشت رشته طولانی و تصادفی هستند که برای افزایش امنیت کوکی‌های ورود کاربران استفاده می‌شوند. آن‌ها یک لایه اضافی از رمزنگاری را به اطلاعات ذخیره‌شده در کوکی‌ها اضافه می‌کنند و حدس زدن آن‌ها را بسیار دشوارتر می‌سازند.

وقتی شما تیک «مرا به خاطر بسپار» را در صفحه ورود می‌زنید، وردپرس اطلاعات ورود شما را در یک کوکی ذخیره می‌کند. اگر این کوکی‌ها به سرقت بروند، هکر می‌تواند به راحتی به حساب شما دسترسی پیدا

کند. Salt Keys با اضافه کردن رشته‌های تصادفی به رمز عبور شما قبل از هش کردن، این فرآیند را بسیار امن‌تر می‌کنند.

وردپرس یک ابزار آنلاین برای تولید این کلیدها ارائه می‌دهد. بسیار مهم است که از کلیدهای منحصر به فرد و تصادفی برای هر سایت استفاده کنید. در صورت مشکوک شدن به هک، یکی از اولین اقدامات باید تغییر این کلیدها باشد. این کار تمام کاربران لاگین‌شده را از سیستم خارج کرده و کوکی‌های آن‌ها را باطل می‌کند.

۶۰. بهترین روش برای حذف کامل یک سایت وردپرسی چیست؟

برای حذف کامل و بدون باقی گذاشتن هیچ اثری از یک سایت وردپرسی، باید دو بخش اصلی را حذف کنید: **فایل‌ها و پایگاه داده**.

1. **حذف فایل‌ها:** با استفاده از FTP یا مدیریت فایل هاست، به ریشه سایت خود بروید و تمام فایل‌ها و پوشه‌های مربوط به نصب وردپرس را انتخاب و حذف کنید. این شامل پوشه‌های `wp-admin`، `wp-content`، `wp-includes` و تمام فایل‌های موجود در ریشه است.

2. **حذف پایگاه داده:** به کنترل پنل هاست خود بروید و وارد `phpMyAdmin` یا بخش مدیریت پایگاه داده شوید. پایگاه داده‌ای که به سایت وردپرسی شما متصل بود را پیدا کنید. روی آن کلیک کرده و گزینه `Drop` را انتخاب کنید. این کار تمام جداول و داده‌های داخل آن را برای همیشه حذف خواهد کرد. همچنین می‌توانید کاربری که به آن پایگاه داده دسترسی داشت را نیز حذف کنید.

هشدار: این عملیات **غیرقابل بازگشت** است. قبل از انجام آن، اطمینان حاصل کنید که از سایت خود یک نسخه پشتیبان کامل و سالم در یک مکان امن دارید، زیرا پس از حذف، هیچ راهی برای بازیابی اطلاعات وجود نخواهد داشت.