

Diagnose and Fix: Debugging WordPress for Site Owners

Facilitator Guide

Overview

When something breaks on a WordPress site, site owners often feel stuck. This 60-minute activity gives participants a repeatable, tool-backed approach to diagnosing and fixing the most common WordPress problems — without needing FTP access, server knowledge, or a developer on call. Participants work entirely in WordPress Playground, so every scenario is safe to try and nothing can break a live site.

Topic	Debugging WordPress for Site Owners
Duration	60 minutes
Audience	Beginner–Intermediate (site owners and editors)
Prerequisite	A modern browser (Chrome, Firefox, Edge, or Safari); no WordPress account required
Environment	WordPress Playground — playground.wordpress.net
Materials	This facilitator guide, slide deck (projected or screen-shared), participants bring their own device

Version note: Steps were verified against WordPress 7.0 in WordPress Playground. WordPress 7.0 changes relevant to this activity: (1) The default admin color scheme is now 'Modern' — a lighter, higher-contrast palette; navigation structure is unchanged. (2) In Settings > General, Administrator and Editor have been removed from the 'New User Default Role' dropdown — this controls the role auto-assigned to self-registered users, not the Add User screen (admins can still assign any role manually). Site Health now alerts if either was set as default before upgrading. (3) WordPress 7.0 requires PHP 7.4 or newer (8.2+ recommended). Check for UI changes before running this activity.

Learning Objectives

By the end of this activity, participants will be able to:

- **Identify** — the most common WordPress error types and what triggers each.
- **Use** — Site Health to proactively surface configuration issues before they become emergencies.
- **Isolate** — plugin and theme conflicts using Troubleshooting Mode — without affecting live visitors.
- **Resolve** — a Critical Error by locating and fixing the root cause using the File Browser.
- **Perform** — a permalink reset and .htaccess regeneration to fix 404 errors.
- **Assess** — which issues you can solve yourself versus when to contact your hosting provider.

Facilitator Setup Notes

Before the session

Open playground.wordpress.net in your own browser and confirm it loads without errors. Test the Health Check & Troubleshooting plugin URL: playground.wordpress.net/?plugin=health-check

Slides

Project the slide deck from the start. Each part has a corresponding slide. Speaker notes carry facilitation cues.

Playground speed

First load can take 10–20 seconds on slower connections. Ask participants to open their Playground tab at the start of each part, not just when needed.

Plugin File Editor

Participants will use Tools > Plugin File Editor in Part 3. Confirm this appears in the Tools menu (it does by default in Playground).

Part 1 · 10 minutes — Site Health — Your Early Warning System

What participants do

1. Open playground.wordpress.net in a new browser tab. WordPress loads automatically.
2. In the left-hand admin menu, go to **Tools > Site Health**.
3. Click the **Status** tab. Read through the results. Notice items marked **Critical** (red), **Recommended** (yellow), and **Good** (green). Ask: what would you do if you saw a Critical item on a live site?
4. Click the **Info** tab. Find the **Copy site info to clipboard** button. This is the information a developer or hosting support team needs when you ask for help — far more useful than a screenshot.

5. Still in Site Health: scroll to the **Persistent object cache** item (or any Recommended item). Read the description and the suggested fix. Note: you do not need to fix it — this is practice reading and interpreting the output.

□ **Facilitator tip:** Site Health was introduced in WordPress 5.2 (2019) and is built into core — no plugin required. It runs automatic background checks and emails site admins about critical issues. Encourage participants to visit it on their own sites this week. Most people have never opened it.

The Info tab is especially valuable for support requests. "Copy site info" outputs structured system data that hosting providers and developers can use immediately, replacing the back-and-forth of basic diagnostics.

Part 2 · 20 minutes — Plugin and Theme Conflicts

Background

Plugin and theme conflicts cause the majority of WordPress errors. The standard diagnostic is to deactivate everything and reactivate one by one. The [Health Check & Troubleshooting plugin](#) makes this safe on a live site: Troubleshooting Mode deactivates all plugins and switches to a default theme **for your browser session only** — other visitors see the site as normal.

What participants do

1. Load Playground with the Health Check plugin pre-installed: playground.wordpress.net/?plugin=health-check. Wait for it to load fully.
2. Go to **Tools > Site Health**. Click the **Troubleshooting** tab (added by the plugin).
3. Click **Enable Troubleshooting Mode**. The page reloads and a green 'Troubleshooting Mode — enabled' notice appears at the top of the screen.
4. Observe: all plugins except Health Check are now deactivated for your session. Go to **Plugins > Installed Plugins** and confirm.
5. You are already on the Troubleshooting Mode management page. In the right panel, expand **Available plugins** and click **Enable** next to each plugin to re-enable them one at a time. This is the isolation workflow: reactivate one plugin, reload, and check whether your issue returns.
6. Click **Disable Troubleshooting Mode** to return the site to its normal state.

□ **Facilitator tip:** The key insight here is session-local isolation. Troubleshooting Mode uses cookies to show a broken or stripped-down admin only to you. It was designed specifically so site owners can diagnose issues on live sites without any risk to visitors.

If time permits: ask participants to think of a recent plugin issue they've had. Would Troubleshooting Mode have helped them narrow it down faster?

Alternative tools to mention: WP_DEBUG (covered in the developer kit) and the Query Monitor plugin both surface conflict information, but they require more technical comfort. Troubleshooting Mode is the right starting point for site owners.

Background

The Critical Error screen ("There has been a critical error on this website") appears when PHP encounters a fatal error. On a real site, WordPress simultaneously sends a recovery email to the admin address with a special one-time Recovery Mode link. In Playground, no email is sent — but we can trigger the error screen and fix it directly using the File Browser, which is actually more instructive than the email flow.

What participants do

1. Go to **Plugins > Installed Plugins** and click **Activate** next to **Hello Dolly**. Hello Dolly must be active for its errors to cause a Critical Error on the front end. Then go to **Tools > Plugin File Editor**.
2. Select **Hello Dolly** from the plugin selector dropdown. The plugin file opens in the editor.
3. At the very top of the file, on line 1, add the text: **this_is_a_syntax_error** (with no PHP comment markers or semicolons). Click **Update File**.
4. Navigate to the front end of the site by clicking **Visit Site** in the admin bar. You should see the Critical Error screen.
5. Go back to the Playground tab. In the top-right Playground toolbar, click the **Sidebar icon** (panel icon). When the panel opens, click the **File browser** tab.
6. In the File Browser, navigate to **wp-content > plugins > hello.php** and click the file to open it.
7. Remove the **this_is_a_syntax_error** text from line 1. The File Browser saves automatically as you type — watch for the brief 'Saving...' indicator next to the file path. Once it disappears, your change is saved.
8. Reload the front end. The Critical Error screen should be gone and the site restored.

□ **Facilitator tip:** On a real site, WordPress also sends a Recovery Mode email to the admin address with a one-time link. That link lets you log in to a stripped-down admin without the fatal plugin active, so you can deactivate it. This is a core WordPress safety feature — added in WordPress 5.2.

The File Browser approach we used here is the equivalent of FTP access, but entirely in the browser. On a real site, participants would use FTP or their hosting file manager to do the same thing.

Key message: fatal errors look scary but they are almost always caused by one plugin with bad code. The fix is usually: deactivate that plugin, or fix the file directly.

What participants do

Work through each quick fix in order. Each takes 1–2 minutes.

Fix 1 — Permalink reset (solves most 404 errors)

1. Go to **Settings > Permalinks**.
2. Without changing anything, click **Save Changes**. This forces WordPress to regenerate the **.htaccess** file and flush rewrite rules.
3. Try navigating to a post on the front end. If a 404 error was caused by stale rewrite rules (e.g., after a migration or server change), this fixes it.

Fix 2 — Browser cache versus live site

Symptom: "I made a change but nothing looks different." This is almost always the browser serving a cached version of the page.

1. Hard refresh: **Cmd+Shift+R (Mac)** or **Ctrl+Shift+R (Windows/Linux)**. This bypasses the browser cache for the current page.
2. If the issue persists on the live site: check whether a caching plugin (e.g., WP Super Cache, W3 Total Cache) is serving stale output. Clear the plugin's cache from the plugin settings page.

Fix 3 — Stuck maintenance mode

*Symptom: "Briefly unavailable for scheduled maintenance. Please check back in a minute." This message appears when a WordPress update fails partway through and leaves a **.maintenance** file in the site root.*

1. Open the Playground **File Browser** (sidebar panel icon in the top-right toolbar).
2. Navigate to the site root (the top level, where **wp-config.php** lives). On a real site, look for a file called **.maintenance**.
3. On a live site: delete the **.maintenance** file via your hosting file manager or FTP. The maintenance message disappears immediately.

When to call your hosting provider

- "Error Establishing a Database Connection" — check wp-config.php credentials first; if correct, the database server may be down.
- "Connection Timed Out" on a shared host — PHP memory or execution time limits; ask the host to increase limits.
- Disk quota exhausted — database shut down by host.
- Server-level errors (50x errors) that persist after plugin/theme deactivation.

□ **Facilitator tip:** These four fixes cover the majority of issues site owners encounter without needing developer help. The key habit to reinforce: always deactivate plugins and switch to a default theme as a first diagnostic step before doing anything else.

When in doubt, the Site Health Info tab gives hosting support everything they need to investigate. Encourage participants to always have that ready.

Debrief — 5 Minutes

1. What was your go-to first step when something broke before today? How does that compare to starting with Site Health or Troubleshooting Mode?
2. Which of the four quick fixes did you not know before? Which one will be most useful for your site?
3. Where does Playground fit into your future troubleshooting workflow? Could you use it to reproduce an issue before contacting your host?

Close by reminding participants: every error in this activity was fixable without FTP, without a developer, and without touching a live site. The same tools and methods work on real WordPress sites.

Sources

Source	URL	Notes
Site Health Screen — Documentation	https://wordpress.org/documentation/article/site-health-screen/	Official documentation for the built-in Tools > Site Health panel.
Common WordPress Errors — Advanced Administration Handbook	https://developer.wordpress.org/advanced-administration/wordpress/common-errors/	Canonical reference for WSOD, Internal Server Error, DB Connection error, Critical Error, PHP errors. Last updated August 2025.
Troubleshooting Using the Health Check — Make WordPress Support Handbook	https://make.wordpress.org/support/handbook/appendix/troubleshooting-using-the-health-check/	Step-by-step guide from the Support team covering Troubleshooting Mode.
Health Check & Troubleshooting Plugin — WordPress.org	https://wordpress.org/plugins/health-check/	Free, open-source plugin. Adds Troubleshooting Mode to Site Health.
Troubleshooting Your Site: Plugin and Theme Conflicts — Learn WordPress	https://learn.wordpress.org/lesson/troubleshooting-your-site-plugin-and-theme-conflicts/	Official Learn WordPress lesson on conflict isolation methodology.
Debugging in WordPress — Advanced Administration Handbook	https://developer.wordpress.org/advanced-administration/debug/debug-wordpress/	Reference for WP_DEBUG constants; used as background for the Critical Error section.
The New WordPress Playground File Browser and UI Enhancements — Make WordPress Playground	https://make.wordpress.org/playground/2025/10/30/the-new-wordpress-playground-file-browser-and-ui-enhancements/	Official announcement confirming File Browser functionality for editing wp-config.php and plugin files. October 2025.