如何检查 WooCommerce 是否已激活

在为 WooCommerce 进行开发时，确保在您的代码运行之前 WooCommerce 已安装并处于激活状态至关重要。 这可以防止与缺少 WooCommerce 函数或类相关的错误。

有几种方法可以实现这一点。 第一种方法是在 `woocommerce_loaded` 动作上执行您的代码。 这种方法可以保证 WooCommerce 及其功能已完全加载并可供使用。 该动作大约在核心 `plugins_loaded` 动作触发的时间点触发。

```
add_action( 'woocommerce_loaded', 'prefix_woocommerce_loaded' );

function prefix_woocommerce_loaded() {
//自定义代码在此处。 WooCommerce 已激活，并且所有插件都已加载...
}
```

注意 ：在此阶段， WordPress 尚未初始化当前用户的相关数据。

另一种方法是在 `woocommerce_init` 动作上执行您的代码。 该动作在 WooCommerce 激活并初始化 之后立即执行。 该动作（以及 `before_woocommerce_init` 动作）在 WordPress 的 `init` 动作的上下文下触发，因此此时当前用户的相关数据已初始化。

```
add_action( 'woocommerce_init', 'prefix_woocommerce_init' );

function prefix_woocommerce_init() {
//自定义代码在此处。 WooCommerce 已激活并初始化...
}
```

注意 ：`before_woocommerce_init` 钩子也是一个选项，它在 WooCommerce 初始化 之前 运行。

使用上述钩子可以访问 WooCommerce 函数，从而可以进行进一步的条件检查。 例如，您可能需要验证 WooCommerce 的版本，以确保与您的代码兼容：

```
add_action( 'woocommerce_init', 'prefix_woocommerce_init' );

function prefix_woocommerce_init() {
//仅当我们可以访问 8.7.0 或更高版本时才继续。
```

```
if ( version_compare( wc()->version, '8.7.0', '<' ) ) {
    return;
}

//自定义代码在此处。 WooCommerce 已激活并初始化...
}
```

根据您的开发需求选择合适的钩子，可以确保您的 WooCommerce 扩展或自定义功能能够平稳高效地运行。

版本号 #2
由 壳壳虫 创建于 2026-03-09 11:53:58 UTC
由 壳壳虫 更新于 2026-03-09 11:53:58 UTC