WP_CLI::add_hook()

目录

- 用法
- 说明
- 相关

安排一个回调函数在特定时机执行。

用法

```
WP_CLI::add_hook( $when, $callback )
```

$when (string) 钩子的标识符。
$callback (callable) 钩子被调用时要执行的回调函数。
@return (void)

说明

钩子在概念上与 WordPress 动作（actions）非常相似。WP-CLI 钩子通常在 WordPress 加载之前调用。

WP-CLI 钩子包括:

- `before_add_command:<command>` - 在命令被添加之前。
- `after_add_command:<command>` - 在命令被添加之后。
- `before_invoke:<command>` (1) - 恰好在命令被调用之前。
- `after_invoke:<command>` (1) - 恰好在命令被调用之后。
- `find_command_to_run_pre` - 恰好在 WP-CLI 找到要运行的命令之前。
- `before_registering_contexts` (1) - 在上下文（contexts）被注册之前。
- `before_wp_load` - 恰好在 WP 加载过程开始之前。
- `before_wp_config_load` - 在 wp-config.php 被定位之后。
- `after_wp_config_load` - 在 wp-config.php 被加载到作用域之后。
- `after_wp_load` - 恰好在 WP 加载过程完成之后。

- `before_run_command` (3) - 恰好在命令被执行之前。

钩子名称后面的括号表示传递给钩子的参数数量。对于此类钩子，回调函数应再次返回第一个参数，使它们像 WP 过滤器（ filter）一样工作。

WP-CLI 命令可以使用 `WP_CLI::do_hook()` 创建自己的钩子。

如果通过 `WP_CLI::do_hook()` 调用传递了额外的参数，这些参数将被传递给 `WP_CLI::add_hook()` 提供的回调函数。

```
# `wp network meta` 确认命令在多站点上下文中执行。
WP_CLI::add_command( 'network meta', 'Network_Meta_Command', array(
   'before_invoke' => function ( $name ) {
       if ( !is_multisite() ) {
           WP_CLI::error( 'This is not a multisite installation.' );
       }
   }
) );
```

内部 *API* 文档在每个版本发布时从 *WP-CLI* 代码库生成。如需改进，请提交一个拉取请求（ *pull request*）。

相关

- **WP_CLI::do_hook()** - 执行注册到给定钩子的回调函数。
- **WP_CLI::add_wp_hook()** - 添加一个回调函数到 WordPress 动作或过滤器。
- **WP_CLI::add_command()** - 向 WP-CLI 注册一个命令。

版本号 #2
由 壳壳虫 创建于 2026-03-26 17:08:46 UTC
由 壳壳虫 更新于 2026-03-26 17:08:46 UTC