Search resources

add_plugins_page(string $page_title, string $menu_title, string $capability, string $menu_slug, callable $callback = '', int $position = null): string|false

Adds a submenu page to the Plugins main menu.

Description

This function takes a capability which will be used to determine whether or not a page is included in the menu.

The function which is hooked in to handle the output of the page must check that the user has the required capability as well.

Parameters

$page_title string required
The text to be displayed in the title tags of the page when the menu is selected.

$menu_title string required
The text to be used for the menu.

$capability string required
The capability required for this menu to be displayed to the user.

$menu_slug string required
The slug name to refer to this menu by (should be unique for this menu).

$callback callable optional
The function to be called to output the content for this page.

Default: ''

$position int optional
The position in the menu order this item should appear.

Default: null

Return

string|false The resulting page's hook_suffix, or false if the user does not have the capability required.

More Information

This function is a simple wrapper for a call to add_submenu_page(), passing the received arguments and specifying 'plugins.php' as the $parent_slug argument. This means the new page will be added as a sub-menu to the Plugins menu.

Source

```
function add_plugins_page( $page_title, $menu_title, $capability, $menu_slug, $callback = '', $posit
    return add_submenu_page( 'plugins.php', $page_title, $menu_title, $capability, $menu_slug, $call
}
```

View all references·View on Trac·View on GitHub

Related

Uses

[add_submenu_page()](#)

wp-admin/includes/plugin.php

Changelog

Version	Description
5.3.0	Added the $position parameter.
3.0.0	Introduced.

User Contributed Notes

Codex 9 years ago   ⌃ 0 ⌄

Basic Plugin Page Example

Typical usage occurs in a function registered with the `admin_menu` hook (see Adding Administration Menus):

```php
/**
 * Add a plugin page.
 */
function wpdocs_plugin_menu() {
    add_plugins_page(
        __( 'WPDocs Plugin Page', 'textdomain' ),
        __( 'WPDocs Plugin', 'textdomain' ),
        'read',
        'wpdocs-unique-identifier',
        'wpdocs_plugin_function'
    );
}
add_action( 'admin_menu', 'wpdocs_plugin_menu' );
```

Log in to add feedback

You must log in before being able to contribute a note or feedback.

About

News

Hosting

Privacy

Showcase

Themes

Plugins

Patterns

Learn

Documentation

Developers

WordPress.tv ↗

Get Involved

Events

Donate ↗

Swag Store ↗

WordPress.com ↗

Matt ↗

bbPress ↗

BuddyPress ↗

CODE IS POETRY