



Search resources



```
add_submenu_page( string $parent_slug, string $page_title, string $menu_title,
string $capability, string $menu_slug, callable $callback = '',
int|float $position = null ): string|false
```

Adds a submenu page.

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

\$parent_slug string required

The slug name for the parent menu (or the file name of a standard WordPress admin page).

\$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 and only include lowercase alphanumeric, dashes, and underscores characters to be compatible with [sanitize_key\(\)](#).

\$callback callable optional

The function to be called to output the content for this page.

Default: ''

\$position int|float 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

Notes

- NOTE: If you're running into the "You do not have sufficient permissions to access this page." message in a `wp_die()` screen, then you've hooked too early. The hook you should use is `admin_menu`.
- For `$menu_slug` please don't use `__FILE__` it makes for an ugly URL, and is a minor security nuisance.
- Within the rendering function `$function` you may want to access parameters you used in `add_submenu_page()`, such as the `$page_title`. Typically, these will work:
 - `$parent_slug: get_admin_page_parent()`
 - `$page_title: get_admin_page_title()`, or simply `global $title`
 - `$menu_slug: global $plugin_page`.
- This function should normally be hooked in with one of the the `admin_menu` actions depending on the menu where the sub menu is to appear:

admin_menu	The normal, or site, administration menu
user_admin_menu	The user administration menu
network_admin_menu	The network administration menu

Source

wp-admin/includes/plugin.php

[Expand code](#) [Copy](#)

```
function add_submenu_page( $parent_slug, $page_title, $menu_title, $capability, $menu_slug, $callback = null ) {
    global $submenu, $menu, $_wp_real_parent_file, $_wp_submenu_nopriv,
           $_registered_pages, $_parent_pages;

    $menu_slug    = plugin_basename( $menu_slug );
    $parent_slug = plugin_basename( $parent_slug );

    if ( isset( $_wp_real_parent_file[ $parent_slug ] ) ) {
        $parent_slug = $_wp_real_parent_file[ $parent_slug ];
    }

    if ( ! current_user_can( $capability ) ) {
        $_wp_submenu_nopriv[ $parent_slug ][ $menu_slug ] = true;
    }
}
```

[View all references](#) · [View on Trac](#) · [View on GitHub](#)

Related

Uses

[get_plugin_page_hookname\(\)](#)

wp-admin/includes/plugin.php

[plugin_basename\(\)](#)

wp-includes/plugin.php

[Show 5 more](#)

Used by

[add_plugin_file_editor_to_tools\(\)](#)

wp-admin/menu.php

[add_management_page\(\)](#)

wp-admin/includes/plugin.php

[add_options_page\(\)](#)

wp-admin/includes/plugin.php

[add_theme_page\(\)](#)

wp-admin/includes/plugin.php

[add_plugins_page\(\)](#)

wp-admin/includes/plugin.php

[Show 9 more](#)

Changelog

Version	Description
5.3.0	Added the <code>\$position</code> parameter.
1.5.0	Introduced.

User Contributed Notes

jakeparis 8 years ago

▲ 97 ▼

Slugs for `$parent_slug` (first parameter)

Dashboard: 'index.php'

Posts: 'edit.php'

Media: 'upload.php'

Pages: 'edit.php?post_type=page'

Comments: 'edit-comments.php'

Custom Post Types: 'edit.php?post_type=your_post_type'

Appearance: 'themes.php'

Plugins: 'plugins.php'

Users: 'users.php'

Tools: 'tools.php'

Settings: 'options-general.php'

Network Settings: 'settings.php'

[Log in to add feedback](#)

Christina Blust 8 years ago

▲ 25 ▼

Adding a submenu page to a custom post type

If you want to add a submenu type to a custom post type, such as a reference page for a custom post type created by a plugin, you can use for the `$parent_slug` parameter whatever you see up top on the “All Posts” view for that post type. For instance, for a custom post type “Book,” the `$parent_slug` could be `'edit.php?post_type=book'`.

Example:

[Expand code](#) [Copy](#)

```
/**  
 * Adds a submenu page under a custom post type parent.  
 */  
function books_register_ref_page() {  
    add_submenu_page(  
        'edit.php?post_type=book',  
        __( 'Books Shortcode Reference', 'textdomain' ),  
        __( 'Shortcode Reference', 'textdomain' ),  
        'manage_options',  
        'books-shortcode-ref',  
        'books_ref_page_callback'  
    );  
}
```

[Log in to add feedback](#)

Codex9 years ago

^ 22 ▾

Inside menu created with [add_menu_page\(\)](#)

If you are attempting to add a submenu page to a menu page created via `add_menu_page()` the first submenu page will be a duplicate of the parent `add_menu_page()`.

If you want a submenu page in this scenario, you should first create a duplicate of your `add_menu_page()` and then add your `add_submenu_page()`:

[Copy](#)

```
add_menu_page('My Custom Page', 'My Custom Page', 'manage_options', 'my-top-level-slug');  
add_submenu_page( 'my-top-level-slug', 'My Custom Page', 'My Custom Page',  
    'manage_options', 'my-top-level-slug');  
add_submenu_page( 'my-top-level-slug', 'My Custom Submenu Page', 'My Custom Submenu Page',  
    'manage_options', 'my-secondary-slug');
```

[Log in to add feedback](#)

Mostafa Soufi7 years ago

^ 19 ▾

Example submenu with php class

[Expand code](#) [Copy](#)

```
/**  
 * Sub menu class
```

```
* @author Mostafa <mostafa.soufi@hotmail.com>
*/
class Sub_menu {

    /**
     * Autoload method
     * @return void
     */
    public function __construct() {
        add_action( 'admin_menu', array(&$this, 'register_submenu') );
    }
}
```

[Show feedback \(1\)](#)[Log in to add feedback](#)

Codex9 years ago

^ 11 ▾

Example

[Expand code](#) [Copy](#)

```
add_action('admin_menu', 'wpdocs_register_my_custom_submenu_page');

function wpdocs_register_my_custom_submenu_page() {
    add_submenu_page(
        'tools.php',
        'My Custom Submenu Page',
        'My Custom Submenu Page',
        'manage_options',
        'my-custom-submenu-page',
        'wpdocs_my_custom_submenu_page_callback' );
}

function wpdocs_my_custom_submenu_page_callback() {
```

To hide your submenu link from a top level menu item to which it belongs you would instead do

[Copy](#)

```
add_action('admin_menu', 'wpdocs_register_my_custom_submenu_page');

function wpdocs_register_my_custom_submenu_page() {
    add_submenu_page(
        null, //or 'options.php'
        'My Custom Submenu Page',
        'My Custom Submenu Page',
        'manage_options',
        'my-custom-submenu-page',
        'my_custom_submenu_page_callback',
    );
}
```

[Show feedback \(2\)](#)[Log in to add feedback](#)

isabel1043 years ago

^ 6 ▾

Regarding the 1st note in the [More Information section above](#), you may also run into the "Sorry, you are not allowed to access this page." message even if you've hooked correctly into the `admin_menu` hook. This error message will happen if all of these are true:

- Your parent page is a custom page that you added using `add_menu_page`, and
- That parent page was created in another plugin, not the same plugin that is creating the submenu page, and
- You didn't add a lower priority to the `admin_menu` hook for the `add_submenu_page` call, so the submenu page may be fired before the parent page—causing an error.

Solution: Add a lower priority, such as 99, to the action that creates the submenu page. For example:

[Copy](#)

```
add_action( 'admin_menu', 'wpdocs_register_my_custom_submenu_page', 99 );
```

[Log in to add feedback](#)

LogixTree7 years ago

^ 4 ▾

When working with the Classes, You can `add_submenu_page` by following Make sure the callable is static function.

[Copy](#)

```
add_submenu_page( 'admin_menu', 'Custom Menu', 'My Custom Menu', 'manage_options', 'my-custom-menu'  
public menu_page_output() {  
    //Menu Page output code  
}
```

[Log in to add feedback](#)

Deepak Rajpal1 year ago

^ 3 ▾

[Copy](#)

```
add_submenu_page(  
    null,  
    'WP Docs Submenu Page',  
    'WP Docs Submenu Page',  
    'manage_options',  
    'wpdocs-submenu-page',  
    'wpdocs_submenu_page_callback',  
) ;
```

In some special cases (like above) if you pass `null` in `parent_slug`, it will generate deprecation warnings when running under PHP 8.1, because the provided `$parent_slug` is passed to `strpos()` and to `str_replace()`. Instead use empty string

if needed like:

[Copy](#)

```
add_submenu_page(  
    '', //or 'options.php'  
    'WP Docs Submenu Page',  
    'WP Docs Submenu Page',  
    'manage_options',  
    'wpdocs-submenu-page',  
    'wpdocs_submenu_page_callback',  
);
```

[Log in to add feedback](#)

Dumitru Brinzan 7 years ago

▲ 2 ▼

To anyone else troubleshooting an unexpected issue with this function, please PAY ATTENTION to the final argument, specifically the \$function, it has to be a STRING, the name of the function, not a call to the function itself. Amateur mistake, I know, but sometimes you just make the simplest of errors.

Bad:

[Copy](#)

```
add_menu_page('My Custom Page', 'My Custom Page', 'manage_options', 'my-top-level-slug', my-output-  
d
```

Good:

[Copy](#)

```
add_menu_page('My Custom Page', 'My Custom Page', 'manage_options', 'my-top-level-slug', 'my-output-  
d
```

[Log in to add feedback](#)

tripflex6 6 years ago

▲ 2 ▼

To further clarify adding a page without it showing in the menu/submenu, using this code:

[Copy](#)

```
add_action('admin_menu', 'wpdocs_register_my_custom_submenu_page');  
  
function wpdocs_register_my_custom_submenu_page() {  
    add_submenu_page(  
        'options.php',  
        'My Custom Submenu Page',  
        'My Custom Submenu Page',  
        'manage_options',  
        'my-custom-submenu-page',  
        'my_custom_submenu_page_callback',  
    );
```

```
);
```

```
}
```

You would then access this page via this URL:

```
/wp-admin/options.php?page=my-custom-submenu-page
```

[Show feedback \(1\)](#)[Log in to add feedback](#)

isMike() 4 years ago

^ 1 ▼

If you are working within a Class, the “function” parameter for the add_submenu_page should be an array, with the first value in the array being an instance of the class while the second value in the array will be a method in the object (given as a string).

Example below

[Expand code](#) [Copy](#)

```
Class WPDocs_AdminPage
{
    private $_plugin_name;

    private $_version;

    public function __construct( $plugin_name, $version )
    {
        $this->_plugin_name = $plugin_name;
        $this->_version = $version;
    }

    public function wpdocs_create_menu_and_submenu_page()
}
```

[Log in to add feedback](#)

rtpHarry 4 years ago

^ 1 ▼

If you're trying to put your custom sub page under a url like:

```
https://www.example.com/wp-admin/admin.php?page=custom-settings
```

Then you should just use the `custom-settings` for the parent slug.

In my use case I wanted to put a custom taxonomy under an ACF Options Page, and I used this:

[Copy](#)

```
add_submenu_page(
    'custom-settings',
    'Product Filter By Ambient Noise',
    'Product Filter By Ambient Noise',
    'manage_options',
    'edit-tags.php?taxonomy=product-filter-ambient-noise'
);
```

[Log in to add feedback](#)

Gerard Reches 4 months ago

0

There is something missing which it is not obvious at all:

When a menu page has 0 or 1 submenu pages, no submenus will show up.

A menu page contains a page by itself.

A menu page that contains 1 submenu page, will appear as the same as if it didn't have a submenu page. This is because the first-level menu element and the first second-level menu belonging to that first-level menu are meant to show the same page.

In other words, you can't have a single submenu element, because that doesn't seem to make sense for WordPress.

"Only one page? Just use the menu element itself then! No need for a dropdown."

Notice that whenever you click on a menu item found in the admin interface that has subitems, it will load its first subitem. So the menu item and its first subitem are essentially the same. However, you are able to rename the subitem.

Some examples of this:

- Dashboard = Home
- Posts = All Posts
- Appearance = Themes
- Tools = Available Tools

When a menu page contains 2 or more submenu pages, the submenu pages will become visible. That's what most of us don't know at first.

So the trick is to create a submenu page identical to the menu page (you can change the \$page_title and \$menu_title if you want) and then your additional submenu page:

[Copy](#)

```
add_menu_page( '', 'Menu Item', $capability, 'parent_slug', $callback );
add_submenu_page( 'parent_slug', 'Page Title', 'First Subitem', $capability, 'parent_slug', $callback );
add_submenu_page( 'parent_slug', 'Second Page Title', 'Second Subitem', $capability, 'submenu_slug' );
```

As you can see, the '\$page_title' for 'add_menu_page' could even be left as an empty string (but not 'null'), because the title used for the page will be the first submenu's '\$page_title'. But I recommend to introduce the same string for readability.

[Log in to add feedback](#)

huntermaster 9 years ago

-7

Parent slugs like 'tools.php' can be found in the old doc

https://codex.wordpress.org/Function_Reference/add_submenu_page#Parameters

[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

