



Search resources



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

Adds a top-level menu 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

\$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 page 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: ''

\$icon_url `string` optional

The URL to the icon to be used for this menu.

- Pass a base64-encoded SVG using a data URI, which will be colored to match the color scheme. This should begin with 'data:image/svg+xml;base64,'.
- Pass the name of a Dashicons helper class to use a font icon, e.g. 'dashicons-chart-pie'.
- Pass 'none' to leave div.wp-menu-image empty so an icon can be added via CSS.

Default: ''

\$position int|float optional

The position in the menu order this item should appear.

Default: null

Return

string The resulting page's hook_suffix.

More Information

Notes

- Important Note: Since WordPress 4.4, you do not need to worry about making the position number unique to avoid conflicts. See trac ticket [#23316](#) for more information.
- If you're running into the "You do not have sufficient permissions to access this page" error, then you've hooked too early. The hook you should use is [admin_menu](#).
- If you only want to move existing admin menu items to different positions, you can use the `admin_menu` hook to unset menu items from their current positions in the global `$menu` and `$submenu` variables (which are arrays), and reset them elsewhere in the array.
- This function takes a 'capability' (see [Roles and Capabilities](#)) 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.
- If you are using the [Settings API](#) to save data, and need the user to be other than the administrator, will need to modify the permissions via the hook `option_page_capability_{$option_group}`, where `$option_group` is the same as `option_group` in [register_setting\(\)](#). Check out the [Settings API](#).

Menu Structure

Default: bottom of menu structure

- 2 – Dashboard
- 4 – Separator
- 5 – Posts
- 10 – Media
- 15 – Links
- 20 – Pages
- 25 – Comments
- 59 – Separator
- 60 – Appearance
- 65 – Plugins
- 70 – Users
- 75 – Tools
- 80 – Settings
- 99 – Separator

For the Network Admin menu, the values are different:

- 2 – Dashboard
- 4 – Separator
- 5 – Sites
- 10 – Users
- 15 – Themes

- 20 – Plugins
- 25 – Settings
- 30 – Updates
- 99 – Separator

Source

wp-admin/includes/plugin.php

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

```
function add_menu_page( $page_title, $menu_title, $capability, $menu_slug, $callback = '', $icon_u
global $menu, $admin_page_hooks, $_registered_pages, $_parent_pages;

$menu_slug = plugin_basename( $menu_slug );

$admin_page_hooks[ $menu_slug ] = sanitize_title( $menu_title );

$hookname = get_plugin_page_hookname( $menu_slug, '' );

if ( ! empty( $callback ) && ! empty( $hookname ) && current_user_can( $capability ) ) {
    add_action( $hookname, $callback );
}
```

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

Related

Uses

[get_plugin_page_hookname\(\)](#)

wp-admin/includes/plugin.php

[sanitize_title\(\)](#)

wp-includes/formatting.php

[set_url_scheme\(\)](#)

wp-includes/link-template.php

[plugin_basename\(\)](#)

wp-includes/plugin.php

[Show 4 more](#)

Used by

[add_object_page\(\)](#)

wp-admin/includes/deprecated.php

[add_utility_page\(\)](#)

wp-admin/includes/deprecated.php

Changelog

Version

Description

[1.5.0](#)

Introduced.

User Contributed Notes

Codex9 years ago

~ 17 ▾

Examples

Add a custom menu item to the WordPress admin menu, for a user with administrator capability:

Method 1:

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

```
/**  
 * Register a custom menu page.  
 */  
function wpdocs_register_my_custom_menu_page() {  
    add_menu_page(  
        __( 'Custom Menu Title', 'textdomain' ),  
        'custom menu',  
        'manage_options',  
        'myplugin/myplugin-admin.php',  
        '',  
        plugins_url( 'myplugin/images/icon.png' ),  
        6  
    );  
}
```

With this method the page-generating code should be located in `myplugin/myplugin-admin.php`:

[Copy](#)

```
<?php esc_html_e( 'Admin Page Test', 'textdomain' ); ?>
```

Method 2:

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

```
/**  
 * Register a custom menu page.  
 */  
function wpdocs_register_my_custom_menu_page(){  
    add_menu_page(  
        __( 'Custom Menu Title', 'textdomain' ),  
        'custom menu',  
        'manage_options',  
        'custompage',  
        'my_custom_menu_page',  
        plugins_url( 'myplugin/images/icon.png' ),  
        6  
    );  
}
```

[Log in to add feedback](#)

Felipe Elia 6 years ago

^ 9 ▼

If you pass a base64-encoded SVG using a data URI, it will only be colored to match the color scheme IF it has a `fill` style attribute. The process is done by `wp-admin/js/svg-painter.js`.

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

surwan 3 years ago

^ 9 ▼

To add a top-level menu item which resolves to a sub-menu item, the menu slugs of each `add_menu_page` and `add_submenu_page` calls need to match. This is useful for when you need the top-level menu item label to be different to the first sub-level item. Otherwise the top-level item label is repeated as the first sub-level item.

Minimal example –

[Copy](#)

```
$menu_slug = 'wpdocs-orders-slug';
add_menu_page( 'WP Docs Orders', 'WP Docs Orders', 'read', $menu_slug, false );
add_submenu_page( $menu_slug, 'Existing WP Docs Orders', 'Existing WP Docs Orders', 'read', $menu_s...
```

In this example; when the 'WP Docs Orders' top-level item is selected, it will trigger the submenu item.

[Log in to add feedback](#)

5Diraptor 2 years ago

^ 6 ▼

Adding custom SVG icons.

I wrestled with this a bit and wanted to add in some notes when using the `$icon_url` parameter. From my experience and from looking on forums, using your custom icon isn't as easy as just creating your SVG and encoding it. Here's the step by step process to add custom SVG icon to the admin menu:

1. First, get or create your SVG icon in any way you want.
2. Next, clean up the SVG file and format it correctly
3. Then base64 encode the file
4. Prefix the base64 string with `'data:image/svg+xml;base64,'`
5. ...and that will give you a valid `$icon_url` string.

So let me expand on those steps.

Step 1

You can create your SVG graphic in any way you want as far as I can see. I created mine in Adobe Illustrator, but anything like Inkscape, Corel draw, even hardcoding it is ok.

Step 2

The painful part is cleaning up your SVG. After a lot of testing and experimenting I found that the cleaner the SVG is, the more likely it is to work. Basically, a lot of graphics programs will add in metadata and other fluff to make the SVG more widely compatible. That's great 99% of the time, but when using it for a wordpress menu icon, it just breaks it.

This is an example of an SVG exported normally from Adobe Illustrator:

Copy

```
<?xml version="1.0" encoding="UTF-8"?><svg xmlns="http://www.w3.org/2000/svg"; viewBox="0 0 15.7 16"><defs><style>.d{fill:#070707;}</style></defs><g id="a"/><g id="b"><g id="c"><path class="d" d="M15.4,6.5H7.9v3h4.4c-.4,2.1-2.3,3.5-4.4,3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/><path class="d" d="M15.4,6.5H7.9v3h4.4c-.4,2.1-2.3,3.5-4.4,3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/></g></g></g></svg>
```

Messy, very messy. Even when we add some line breaks and indent it, there's a lot going on.

Expand code Copy

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg"; viewBox="0 0 15.7 16">
  <defs>
    <style>
      .d{fill:#070707;}</style>
  </defs>
  <g id="a"/>
  <g id="b">
    <g id="c">
      <path class="d" d="M15.4,6.5H7.9v3h4.4c-.4,2.1-2.3,3.5-4.4,3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/>
      <path class="d" d="M15.4,6.5H7.9v3h4.4c-.4,2.1-2.3,3.5-4.4,3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/>
    </g>
  </g>
</g></svg>
```

So what we want to do is clean this up a lot and get rid of unnecessary data. You may find that the above will load if you add it as an image source but it won't color properly with the GUI and it will probably be the wrong size. So there's a few things we need to do:

- Remove the `<?xml` tag, this won't help
- Then you will have to hardcode each style into the path, rather than using the `style` tag. For each path, work out which classes apply to the path from the tag, and add the CSS from those classes into the path.
- Once that's done, you can remove the `defs` element completely.
- You can also remove the elements and leave the elements at the root of the SVG.
- To ensure your SVG is colored in line with the GUI, you will need to add `fill="black"` to each path.
- And then, to ensure your SVG is the right width and height, add `width="20" height="20"` into the opening element

Once that's done, you'll end up with an SVG file that looks more like this:

Copy

```
<svg width="20" height="20" viewBox="0 0 20 20" xmlns="http://www.w3.org/2000/svg">
  <path fill="black" d="M17.6 8.5h-7.5v3h4.4c-.4 2.1-2.3 3.5-4.4 3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/>
  <path fill="black" d="M17.6 8.5h-7.5v3h4.4c-.4 2.1-2.3 3.5-4.4 3.4-2.6-.1-4.6-2.1-4.7-4.7-.1-2.7,2-5,4.7-5.1,1.1,0,2.2,.4,3.1,1.212.3-2.2c-1.4-1.4-3.4-2.1-5.3-2.1C3.6,0,0,3.6,0,8s3.6,8,8,8,7.7-3.2,7.7-7.8c-.1-.6-.1-1.1-.3-1.7Z"/>
</svg>
```

```
</svg>
```

Step 3

Now you want to base64 encode your SVG. The way I did it was to pass the file into PHP's default encoding mechanism, and copy the path into my PHP as an absolute path. You don't want PHP to read the SVG file, base64 encode it, and pass it to the menu every time the user reloads the page, that's just a waste of resources and time.

[Copy](#)

```
// load the SVG data by loading the file or including the XML directly.  
$svg = file_get_contents( '/path/to/icon.svg' );  
// or  
$svg = '<svg width="20" height="20" viewBox="0 0 20 20" xmlns="http://www.w3.org/2000/svg"><path fi...  
  
// then encode it and echo it out  
echo base64_encode( $svg );  
// will give you a long string that looks like this:  
// PHN2ZyB3aWR0aD0iMjAiIGHlaWdodD0iMjAiIHZpZXdib3g9IjAgMCAYMCAYMCAYMCAYG1sbnM9Imh0dHA6Ly93d3cudzMub3Jn...  
  
// that's your base64 encoded SVG file!
```

Step 4

Add 'data:image/svg+xml;' in front of the base64 SVG so it looks like this:

[Copy](#)

```
"...
```

and there, sir, is a perfectly valid `$icon_url` string. I would hard code that string to a variable and simply add it into your plugin every time.

[Copy](#)

```
add_menu_page( ___( 'WPDocs Plugin page name' ), ___( 'WPDocs Plugin' ), 'manage_options', 'slug', 'ca...
```

I hope that helps someone else, because that would have saved me several hours this morning!

[Log in to add feedback](#)

Sophy 7 years ago

4

Add Menu Pages with Dashicons. More dashicons check here <https://developer.wordpress.org/resource/dashicons/#menu>

[Copy](#)

```
add_action( 'admin_menu', 'register_my_custom_menu_page' );  
function register_my_custom_menu_page() {  
    // add_menu_page( $page_title, $menu_title, $capability, $menu_slug, $function, $icon_url, $posit...  
    add_menu_page( 'Custom Menu Page Title', 'Custom Menu Page', 'manage_options', 'custom.php', '' );  
}
```

[Log in to add feedback](#)

Aamer Shahzad8 years ago

^ 3 ▾

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

```
/**  
 * Create admin Page to list unsubscribed emails.  
 */  
// Hook for adding admin menus  
add_action('admin_menu', 'wpdocs_unsub_add_pages');  
  
// action function for above hook  
  
/**  
 * Adds a new top-level page to the administration menu.  
 */  
function wpdocs_unsub_add_pages() {  
    add_menu_page(  
        . . . . .
```

[Log in to add feedback](#)

eliran4 years ago

^ 2 ▾

A better example to peskyesky.

object oriented example.

TODO:

create includes/layout.php

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

```
class WPDocs_EB_EbtechModules {  
    public static function init() {  
        add_action( 'admin_enqueue_scripts', array( __CLASS__, 'adminAssets' ) );  
        add_action( 'admin_menu', array( __CLASS__, 'adminMenu' ) );  
    }  
  
    public static function adminMenu() {  
        add_menu_page(  
            __( 'WPDocs Ebtech Menu', 'wpdocs-webnail-modules' ),  
            __( 'WPDocs Ebtech modules', 'wpdocs-webnail-modules' ),  
            'manage_options',  
            'wpdocs-ebtech-modules',  
            array( __CLASS__, 'menuPage' ),
```

[Log in to add feedback](#)

Vincent Dubroeucq 4 years ago

^ 0 ▾

The callback function used to display the contents of the page isn't necessarily required to check current user's capabilities, since the callback is not even hooked in if the user doesn't have the right capabilities.

If you try to directly access a page for which you don't have required capabilities, you'll get an error page.

[Log in to add feedback](#)

angelbouklis 9 months ago

^ 0 ▾

Quick example on how to add a menu entry with an icon.

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

```
<?php

add_action( 'admin_menu', 'wpdocs_add_artist_menu' );
function wpdocs_add_artist_menu(): void {
    add_menu_page(
        'Artists',
        'Artists',
        'manage_options',
        'admin-artists',
        '...',
        'data:image/svg+xml;base64,' . base64_encode(
            'svg goes here, unfortunately I can't paste one because it is automatically removed'
        )
    )
}
```

[Log in to add feedback](#)

peskyesky 8 years ago

^ -5 ▾

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

```
// For those who are object orientated. Add a class
// function as the menu callback and setup the
// menus automatically.

// Exit if accessed directly
if ( !defined( 'ABSPATH' ) ) exit;

class MyMenuSetterUpper {

    private static $instance;

    /**
     * Main Instance

```

[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

