

Admin

Hooks in this category let you extend and customize the WordPress admin area — from initialization to menus, notices, footer text and more.

Control the admin area, add menus, notices, and more.



admin_init

Fires after WordPress initializes the admin area.

```
<?php
add_action(
    'admin_init',
    'my_admin_init'
);
```

01

Admin
initialization



wp_mail_from

Filters the sender email address for outgoing emails.

```
<?php
add_filter(
    'wp_mail_from',
    'my_mail_from'
);
```

02

Email
sender



admin_footer_text

Filters the text shown in the admin footer.

```
<?php
add_filter(
    'admin_footer_text',
    'my_footer_text'
);
```

03

Footer
text



admin_menu

Fires when the admin menu is being built.

```
<?php
add_action(
    'admin_menu',
    'my_admin_menu'
);
```

04

Admin
menu



admin_notices

Fires when admin notices are displayed.

```
<?php
add_action(
    'admin_notices',
    'my_admin_notices'
);
```

05

Notices



manage_posts_columns

Filters the columns shown in the posts list table.

```
<?php
add_filter(
    'manage_posts_columns',
    'my_posts_columns'
);
```

06

Posts list
columns

4WP

4wp.dev/hooks/category/admin/

Extend. Customize. Manage.

Use these hooks to take full control of the WordPress admin area.

admin_init()

The `admin_init` action hook is fired immediately after the WordPress `admin` area is initialized. It runs *before* any admin page is rendered.

*Fired after the admin environment is set up but *before* any admin page loads.*



```
add_action( 'admin_init',  
    'my_register_settings' );
```

Register Settings



```
add_action( 'admin_init',  
    'my_restrict_access' );
```

Restrict Access



```
add_action( 'admin_init',  
    'my_load_textdomain' );
```

Load Textdomain



```
add_action( 'admin_init',  
    'my_admin_notices' );
```

Show Admin Notices

```
<?php  
add_action( 'admin_init',  
    'my_admin_init_setup' );  
  
function my_admin_init_setup() {  
    // Your code here.  
  
    // Examples:  
    // register_setting( ... );  
    // add_rewrite_rule( ... );  
    // check_admin_referer( ... );  
}  
?>
```

Initialize the admin area.

Use `admin_init` to register settings, load resources, restrict access, and prepare the admin experience.

WORDPRESS FILTERS

wp_mail_from

The wp_mail_from filter lets you change the sender email address used by WordPress.



Change sender address

Use a custom email address instead of the default one.



Use a real domain

Improve deliverability and avoid spam issues.



Brand outgoing mail

Make your emails look more professional.



Route email by type

Send different emails from different addresses.



Change the sender address with the **<filter>** hook.

```
<?php
add_filter( 'wp_mail_from',
    function() {
        return 'no-reply@example.com';
    } );
```

4WP

4wp.dev/hooks/filters/wp_mail_from

Power your **<filter>**.

Customize the sender email address for WordPress email.

WORDPRESS FILTERS

admin_footer_text

The `admin_footer_text` filter lets you customize the footer text shown in WordPress admin.



White-label the footer

Replace the default WordPress credit.



Add a support link

Give clients quick access to help.



Show your version

Display plugin or theme information.



Hide footer text

Return an empty string.



Customize the admin footer with the `<filter>` hook.

```
<?php
add_filter( 'admin_footer_text',
    function() {
        return 'Managed by Acme Agency -
        <a href="https://example.com/support">
        Get support</a>';
    }
);
```

4WP

4wp.dev/hooks/filters/admin_footer_text

Power your `<filter>`.

Customize the WordPress admin footer with the `admin_footer_text` filter.

admin_menu()

The `admin_menu` action hook is fired when the admin menu is being built in the WordPress backend `<wp-admin>`.



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

Add a custom menu item



```
add_menu_page(  
    'My Plugin', 'My Plugin',  
    'manage_options', 'my-plugin',  
    'my_plugin_page' );
```

Create a top-level menu



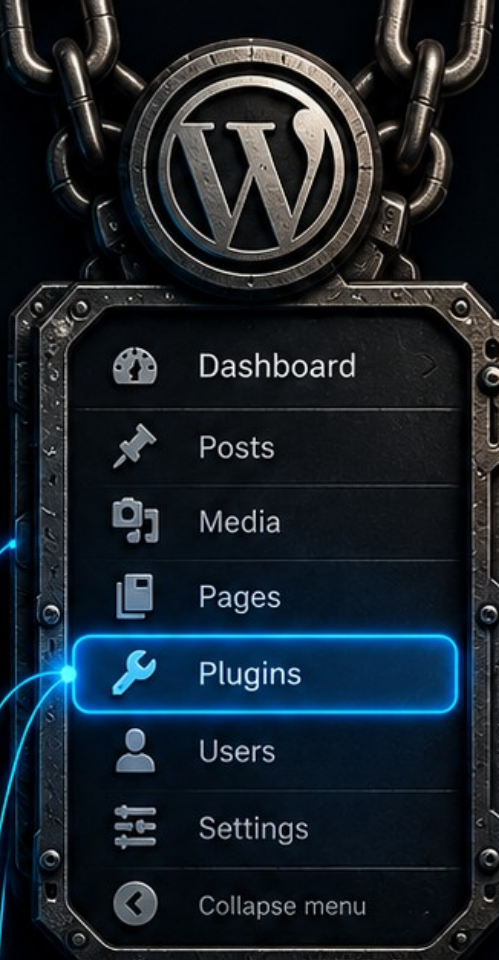
```
add_submenu_page(  
    'my-plugin', 'Settings',  
    'Settings', 'manage_options',  
    'my-plugin-settings', 'my_plugin_settings' );
```

Add a submenu page



```
add_menu_page(  
    'Settings', 'Settings',  
    'manage_options', 'options-general.php',  
    'options-general.php' );
```

Manage existing menu pages



```
<ul class="wp-menu">  
    <li class="wp-first-item">  
        <a href="edit.php?post_type=post">  
            <span class="dashicons dashicons-admin-post"></span>  
            Posts  
        </a>  
    </li>  
    <li class="wp-menu-separator"></li>  
    ...  
</ul>
```



4wp.dev/hooks/actions/admin_menu/

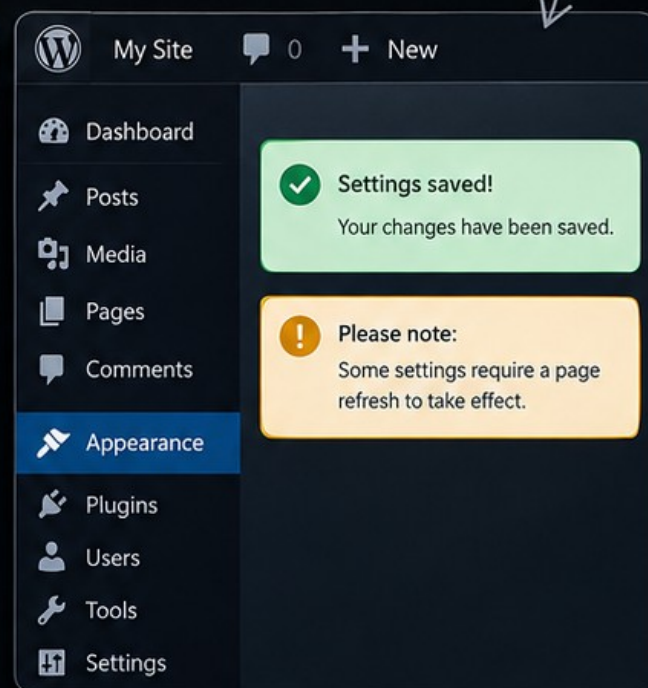
Extend your WordPress admin.

Use the `admin_menu` action hook to add and customize menu items in the backend.

admin_notices()

The `admin_notices` action hook is fired when WordPress is ready to display admin notices in the backend. `<admin_notices>` It's the perfect place to add custom notices to the admin area.

Admin notices are displayed in the backend (notice area)



```
add_action( 'admin_notices',  
    'my_custom_notice' );
```

Add a custom admin notice



```
add_action( 'admin_notices',  
    'my_info_notice' );
```

Add an informational notice



```
add_action( 'admin_notices',  
    'my_warning_notice' );
```

Add a warning notice



```
add_action( 'admin_notices',  
    'my_error_notice' );
```

Add an error notice

```
<?php  
function my_custom_notice() {  
    echo '<div class="notice notice-success">';  
    echo '<p>Custom admin notice goes here!</p>';  
    echo '</div>';  
}  
  
add_action( 'admin_notices', 'my_custom_notice' );  
?>
```



4wp.dev/hooks/actions/admin_notices/

Show notices. Inform users.

Use the `admin_notices` action hook to add custom messages to the WordPress admin area.

WORDPRESS FILTERS

manage_posts_columns



The `manage_posts_columns` filter lets you customize the columns shown in WordPress post lists.



Customize post list columns with the `<filter>` hook.



Add custom columns

Show your own data in the post list.



Remove columns

Hide unnecessary fields.



Rename columns

Change column labels in the admin table.



Reorder columns

Control the admin table layout.

```
<?php
add_filter( 'manage_posts_columns',
    function( $columns ) {
        $columns['custom'] = 'Custom';
        return $columns;
    } );
} );
```



4wp.dev/hooks/filters/manage_posts_columns

Power your `<filter>`.

Customize WordPress post list columns.