Sidebars

- What are Sidebars
- Registering a Sidebar
- Examples
- Displaying Sidebars in your Theme
  - Create a Sidebar Template File
  - Example:
  - Load your Sidebar
- Customizing your Sidebar
  - Display Default Sidebar Content
  - Display Default Widgets

What are Sidebars

A sidebar is any widgetized area of your theme. Widget areas are places in your theme where users can add their own widgets. You do not need to include a sidebar in your theme, but including a sidebar means users can add content to the widget areas through the Customizer or the Widgets Admin Panel.

Widgets can be employed for a variety of purposes, ranging from listing recent posts to conducting live chats.

The name “sidebars” comes from a time when widget areas were normally created in a long strip to the lefthand or righthand side of a blog. Today, sidebars have evolved beyond their original name. They can be included anywhere on your website. Think of a sidebar as **any area that contains widgets**.

Registering a Sidebar

To use sidebars, you must register them in `functions.php`.

To begin, `register_sidebar()` has several parameters that should always be defined regardless of whether they are marked as optional. These include x, y, and z.

- **name** – your name for the sidebar. This is the name users will see in the Widgets panel.
- **id** – must be lowercase. You will call this in your theme using the `dynamic_sidebar` function.
- **description** – A description of the sidebar. This will also be shown in the admin Widgets panel.
- **class** – The CSS class name to assign to the widget's HTML.
- **before_widget** – HTML that is placed before every widget.
- **after_widget** – HTML that is placed after every widget. Should be used to close tags from `before_widget`.
- **before_title** – HTML that is placed before the title of each widget, such as a header tag.
- **after_title** – HTML that is placed after every title. Should be used to close tags from `before_title`.

To register a sidebar we use `register_sidebar` and the `widgets_init` function.

```
<?php
function themename_widgets_init() {
register_sidebar( array(
'name'          => __( 'Primary Sidebar', 'theme_name' ),
'id'            => 'sidebar-1',
'before_widget' => '<aside id="%1$s" class="widget %2$s">',
'after_widget'  => '</aside>',
'before_title'  => '<h3 class="widget-title">',
'after_title'   => '</h3>',
) );
register_sidebar( array(
'name'          => __( 'Secondary Sidebar', 'theme_name' ),
'id'            => 'sidebar-2',
'before_widget' => '<ul><li id="%1$s" class="widget %2$s">',
'after_widget'  => '</li></ul>',
'before_title'  => '<h3 class="widget-title">',
'after_title'   => '</h3>',
) );
}
```

Registering a sidebar tells WordPress that you're creating a new widget area in **Appearance > Widgets** that users can drag their widgets to. There are two functions for registering sidebars:

- register_sidebar()
- register_sidebars()

The first lets you register one sidebar and the second lets you register multiple sidebars.

It is recommended that you register sidebars individually as it allows you to give unique and descriptive names to each sidebar.

Examples

For widget areas in header and footer, it makes sense to name them “Header Widget Area” and “Footer Widget Area”, rather than “Sidebar 1” and “Sidebar 2” (which is the default).
This provides a useful description of where the sidebar is located.

The following code, added to `functions.php` registers a sidebar:

```
<?php
add_action( 'widgets_init', 'my_register_sidebars' );
function my_register_sidebars() {
	/* Register the 'primary' sidebar. */
	register_sidebar(
		array(
			'id'            => 'primary',
			'name'          => __( 'Primary Sidebar' ),
			'description'   => __( 'A short description of the sidebar.' ),
			'before_widget' => '<div id="%1$s" class="widget %2$s">',
			'after_widget'  => '</div>',
			'before_title'  => '<h3 class="widget-title">',
			'after_title'   => '</h3>',
		)
	);
	/* Repeat register_sidebar() code for additional sidebars. */
}
```

The code does the following:

- `register_sidebar` – tells WordPress that you’re registering a sidebar
- `'name' => __( 'Primary Widget Area', 'mytheme' ),` – is the widget area’s name that will appear in Appearance > Widgets
- `'id' => 'sidebar-1'` – assigns an ID to the sidebar. WordPress uses ‘id’ to assign widgets to a specific sidebar.
- `before_widget` / `after_widget` – a wrapper element for widgets assigned to the sidebar. The “%1$s” and “%2$s” should always be left in `id` and `class` respectively so that plugins can make use of them. By default, WordPress sets these as list items but in the above

example they have been altered to div.
- `before_title` / `after_title` – the wrapper elements for the widget's title. By default, WordPress sets it to h2 but using h3 makes it more semantic.

Once your sidebar is registered, you can display it in your theme.

Displaying Sidebars in your Theme

Now that your sidebars are registered, you will want to display them in your theme. To do this, there are two steps:

1. Create the `sidebar.php` template file and display the sidebar using the `dynamic_sidebar` function
2. Load in your theme using the `get_sidebar` function

Create a Sidebar Template File

A sidebar template contains the code for your sidebar. WordPress recognizes the file `sidebar.php` and any template file with the name `sidebar-{name}.php`. This means that you can organize your files with each sidebar in its own template file.

Example:

1. Create `sidebar-primary.php`
2. Add the following code:

```
<div id="sidebar-primary" class="sidebar">
<?php dynamic_sidebar( 'primary' ); ?>
</div>
```

Note that `dynamic_sidebar` takes a single parameter of `$index`, which can be either the sidebar's name or id.

Load your Sidebar

To load your sidebar in your theme, use the `get_sidebar` function. This should be inserted into the template file where you want the sidebar to display. To load the default `sidebar.php` use the following:

```
<?php get_sidebar(); ?>
```

To display the Primary sidebar, pass the `$name` parameter to the function:

```
<?php get_sidebar( 'primary' ); ?>
```

Customizing your Sidebar

There are lots of ways that you can customize your sidebars. Here are some examples:

Display Default Sidebar Content

You may wish to display content if the user hasn't added any widgets to the sidebar yet. To do this, you use the `is_sidebar_active()` function to check to see if the sidebar has any widgets. This accepts the `$index` parameter which should be the ID of the sidebar that you wish to check.

This code checks to see if the sidebar is active, if not it displays some content:

```
<div id="sidebar-primary" class="sidebar">
	<?php if ( is_active_sidebar( 'primary' ) ) : ?>
		<?php dynamic_sidebar( 'primary' ); ?>
	<?php else : ?>
		<!-- Time to add some widgets! -->
	<?php endif; ?>
</div>
```

Display Default Widgets

You may want your sidebar to be populated with some widgets by default. For example, display the Search, Archive, and Meta Widgets. To do this you would use:

```
<div id="primary" class="sidebar">

	<?php do_action( 'before_sidebar' ); ?>

	<?php if ( ! dynamic_sidebar( 'sidebar-primary' ) ) : ?>

		<aside id="search" class="widget widget_search">
```

```
□□□<?php get_search_form(); ?>
□□</aside><!-- #search -->

□□<aside id="archives" class"widget">
□□□<h3 class="widget-title"><?php _e( 'Archives', 'shape' ); ?></h3>
□□□<ul>
□□□□<?php wp_get_archives( array( 'type' => 'monthly' ) ); ?>
□□□</ul>
□□</aside><!-- #archives -->

□□<aside id="meta" class="widget">
□□□<h3 class="widget-title"><?php _e( 'Meta', 'shape' ); ?></h3>
□□□<ul>
□□□□<?php wp_register(); ?>
□□□□<li><?php wp_loginout(); ?></li>
□□□□<?php wp_meta(); ?>
□□□</ul>
□□</aside><!-- #meta -->

□<?php endif; ?>

</div><!-- #primary -->
```

□□□ #2
□ □□□ □□□ 2026-08-29 15:36:32 UTC
□ □□□ □□□ 2026-08-29 15:36:32 UTC