I'm not robot
reCAPTCHA
Continue

Woocommerce single product page sidebar

Remove sidebar from single product page woocommerce. Add sidebar to woocommerce single product page.

Need help with your WordPress project? Hire an experienced developer today! WooCommerce can be a bit âdifficult when it comes to placing the WordPress sidebar and viewing. If you’re looking for a way to disable the WooCommerce sidebar and make your Woocommerce pages full-width, get ready to do some tweaking! We’ll take care of the right and wrong ways to disable or remove the sidebar from your Woo pages, and make your store awesome! But first, let’s see how you can activate the sidebar in case you screwed up. Using one of our Themes? If you use one of our Podcast themes, such as the Tusant WordPress Theme, you can easily change the WooCommerce settings on the page you use as the store page under the “Options Page” panel. In addition, you will find a number of additional settings in the “Look > Customize” screen.Enable Sidebar The WordPress sidebar should be enabled by default. Otherwise, you may need to go to Dashboard “Appearance” Widgets, and add some widgets to the sidebar to make it visible. WooCommerce comes with custom widgets that you can select called “Show Overview Page” and “Single Product Page”. Just drag and drop them in the sidebar, and they should appear in your store. Also, you may need to go to the page in question, and select a template with a sidebar. This can happen when you have a Shop page that uses a full-width template (without a sidebar). Disable the WooCommerce sidebar There are a couple of different options you can use to disable the WooCommerce sidebar. We’ll go beyond all the options and show you how to disable the WooCommerce sidebar yourself: The “wrong” way There’s a good chance you read somewhere that the easiest way to remove the sidebar is to use the following lines of CSS: .single-product .sidebar { display: none !important; }, or even something like: .woocommerce #sidebar { display: none; } The first snippet will hide the sidebar from individual product pages, while the second one will hide it completely on every WooCommerce page on your WordPress site. But the trick here is that even if you won’t see the sidebar, it will still be rendered and take up space in your template. This is because CSS cannot prevent an item from being displayed, but can simply hide it. Also, this would not necessarily work for all themes, and if you’re using a custom theme it might not work properly, so make sure you check it again. The right way to disable the WooCommerce sidebar To properly disable the WooCommerce sidebar, You need to write some PHP code, and manually change what WordPress should make it on the front end. Here are a couple of solutions: 1. Disable the sidebar Use a WordPress hook (recommended method) This is the best solution, and is considered the «WordPress way» to do things. Browse the Functions.php file of the theme and write the following code: Function disable_woo_commerce_sidebar () {remove_action ((âwoocommerce_get_sidebar', 10); } add_action (âinit', âdisable_woo_commerce_sidebar'); 2. Disabling the sidebar in a custom template that displays WooCommerce products next to regular WordPress messages You can use the is_woocommerce () function to detect whether the current page you are accessing is a WooCommerce template or not. The function will return true if it is a Woo page, and false otherwise. So in the sidebar.php of our theme, we could write: if (!is_woocommerce ()) { get_sidebar (); } Using this logic, we can easily determine what kind of page the user is accessing, and based on that, we can call up the function displaying the sidebar. So let’s say you open a product page (which is a WooCommerce template). Then the expression inside the if statement will return false, and the sidebar function will not be called “” which means the sidebar will not be displayed. How does !is_woocommerce () return false? Well, the is_woocommerce () function will return true, and the ! operator (not) will convert the “true” to the opposite value, which is false. 3. Disabling the sidebar in a WooCommerce template Switch to woocommerce/templates/shop, and open the file sidebar.php. Here, you need to comment everything in the file sidebar.php using /* at the top, and */ at the bottom of the file. Save it and see if it worked on the front end. If you need an experienced WordPress developer to help you, please consider hiring an expert. Hi guys, When I was looking for a “how to do”, I found a lot of topics, but all for the old versions of “X” and they didn’t work. What code should I add to my child’s theme feature file to add a sidebar to the individual product page in WooCommerce, with the latest version of X and WC? Thanks so much! This response has been flagged as private. Hello There, Thanks for writing in! Since this requires a change in the template, I would suggest setting a child theme. This allows you to make changes to the code that will not be overwritten when an X update is released. After the child theme has been set up, please review how it is recommended to make changes to the template in Customization Best Practices. After that, add the following to the child theme’s functions.php file: if (! function_exists (âx_get_content_layout')): function x_get_content_layout () { $stack = x_get_stack (); $content_layout = x_get_option (âx_ ' . $stack . â_layout_content', âcontent', âcontent', âcontent', âcontent'-sidebar'); if ($content_layout != âfull-width') { if (is_home ()) { $opt = x_get_option (âx_blog_layout', âsidebar'); $layout = ($opt == âsidebar')? $content_layout: $opt; } elseif (is_singular) (âpost ')) { $meta = get_post_meta (get_the_ID (), â_x_post_layout', true); $layout = == âon')? âfull-width': $content_layout; } elseif (x_is_portfolio_item ()) { $layout = âfull-width'; } else elseif (x_is_portfolio ()) { $meta = get_post_meta (get_the_ID (), â_x_portfolio_layout', true); $layout = ($meta == âsidebar')? $content_layout: $meta; } elseif (is_page_template (( ) Ï Ï Ï