Decimal Quantities Causing Errors in Cart or Checkout

Troubleshooting · https://wpfactory.com/docs/decimal-quantity-errors/

If quantities from Min Max Step Quantity Limits Manager for WooCommerce cause errors in the block-based Cart or Checkout, turn on **Woocommerce block compatibility** at **WPFactory > Quantity Manager** (also reachable at **WooCommerce > Settings > Product Quantity**) > **Advanced**. For true decimal quantities, use the classic cart and checkout shortcodes.

WooCommerce's block-based Cart and Checkout expect whole-number quantities. The steps below cover both the type errors you can fix with a setting and the decimal limits of the blocks.

Common symptoms

- A fatal error such as "DivisionByZeroError: Modulo by zero" in a WooCommerce core file when decimals are used.
- Decimal input rounding down to a whole number (1.5 becomes 1) after a cart update or navigation.
- Decimals accepted on the product page but reset to whole numbers, or failing, at cart or checkout.
- An error such as "Unexpected error in: woocommerce/cart-line-items-block" with "You must provide an integer", or a PHP error saying `QuantityLimits::limit_to_multiple(): Argument #1 ($number) must be of type int, string given`.

Cause

WooCommerce's block-based Cart and Checkout do not reliably support decimal quantities yet. The quantity normalization the blocks perform assumes whole numbers, which can trigger a division or modulo error. This is a known WooCommerce limitation, not a fault in your settings.

Fix

1. **Turn on Woocommerce block compatibility.** Go to **WPFactory > Quantity Manager**, open the **Advanced** section, tick **Woocommerce block compatibility** (`wpfmmsq_advance_wc_block_api`, off by default) and save. Quantity Manager then passes your minimum, maximum and step to the blocks as whole numbers instead of text values (a minimum below 1 becomes 1), which removes the "must be of type int" and "You must provide an integer" errors. With **Decimal quantities** on, a value that really has decimals, such as a step of `0.5`, is still passed as a decimal, so the blocks' decimal limits below still apply to those products.
2. **Switch Cart and Checkout to the classic (shortcode) versions.** Edit your Cart and Checkout pages so they use the `[woocommerce_cart]` and `[woocommerce_checkout]` shortcodes instead of the block templates. Decimals work reliably in classic mode.
3. **Test in isolation.** Switch to a default theme with only WooCommerce and Quantity Manager active to rule out theme or plugin conflicts.
4. **Update to the latest version.** Make sure Quantity Manager and WooCommerce are current.
5. **If you must keep block Cart/Checkout,** use whole-number quantities until WooCommerce's decimal support in blocks stabilizes.

Product page block does not show the minimum or step

If your single product template uses WooCommerce's **Add to Cart + Options (Beta)** block and its quantity box ignores your minimum or step, test the product page with the **Add to Cart with Options** block instead. That block renders WooCommerce's standard add-to-cart form, and Quantity Manager applies its limits through the standard quantity field arguments. Cart and checkout still check the rules either way.

> **Note:** reverting to the classic Cart and Checkout is currently the most reliable workaround for decimal support.

How to confirm it is fixed

Add a product with a minimum or step rule to the cart, open the block Cart, change the quantity with the stepper, and continue to the block Checkout. No error should appear, and the quantity should move in your step.

Still stuck? Our team answers every ticket on the Quantity Manager support forum.



QUANTITY MANAGER

See every feature and the pricing on the Quantity Manager plugin page.

View plugin →

Related articles

- How to Block Checkout When Cart Quantities Are Wrong
- Stock Update Fails with Decimal Quantities
- Cart or Order Page Fails with Decimal Quantity Products
- Customers Can Still Add Multiple Quantities to the Cart
- How to Validate and Auto-Correct Cart Quantities