Additional Checkout Fields: Developer Hooks and Filters Reference

Developers · https://wpfactory.com/docs/custom-checkout-fields-hooks-filters/

Developer hooks and filters reference

Custom Checkout Fields for WooCommerce exposes a small, stable set of filters and helper functions so you can change field behavior, labels, visibility, and fees in code. Every filter listed here is called from the plugin's own source in version 1.9.5. There are no deprecated hook prefixes to migrate.

Core filters

Filter	Filtered value	Arguments
`alg_wc_ccf_total_fields`	Number of fields the plugin renders (default `1`)	`$value`
`alg_wc_ccf_get_field_option`	A single field option value as read from the database	`$result`, `$option`, `$field_nr`, `$default`, `$context`
`alg_wc_ccf_field_label`	The label shown on the checkout field	`$label, $data, $field_nr`

Filter	Filtered value	Arguments
`alg_wc_ccf_order_field_label`	The label shown on order, email, and thank-you output	`$label`, `$field_data`
`alg_wc_ccf_admin_order_field_label`	The label shown on the admin order edit screen	`$label`, `$field_data`
`alg_wc_ccf_field_visible`	Whether a field is shown at checkout (boolean)	`$visible`, `$field_nr`, `$frontend`
`alg_wc_ccf_fee_cart_total`	The cart total a percent fee is calculated from	`$total`, `$field_nr`
`alg_wc_ccf_min_max_cart_amount_cart_total`	The cart total tested by Min/Max cart amount visibility	`$cart_total`, `$field_nr`
`alg_wc_custom_checkout_fields_always_visible_on_empty_cart`	Render fields when the cart is empty (default `false`)	`$value`

Filter	Filtered value	Arguments
`alg_wc_ccf_settings`	Settings-screen markup and attributes; the Pro version uses it to raise the field count and strip the upsell copy	`$value, $type`

Helper functions

Function	Returns
`alg_wc_ccf_get_option( $option, $default )`	A general plugin option (reads `alg_wc_ccf_<option>`).
`alg_wc_ccf_get_field_option( $option, $field_nr, $default, $context )`	A per-field option (reads `alg_wc_ccf_<option>_<n>`).
`alg_wc_ccf_get_order_fields_data( $order_id )`	The order's consolidated field data array.
`alg_wc_ccf_update_order_fields_data( $order_id, $data )`	Writes the field data array back to the order.

Shortcodes

Three shortcodes ship with the plugin. They are meant for use inside field options (such as a default value or a datepicker min/max date) and in content.

Shortcode	Purpose
`[alg_wc_ccf_datetime]`	Outputs the current date/time as a formatted string. Accepts a `format` attribute.

Shortcode	Purpose
`[alg_wc_ccf_if]`	Conditional output. Compares `value1` and `value2` with an `operator` and returns `then` or `else`.
`[alg_wc_ccf_translate]`	Outputs different text per language, keyed on the WPML language.

Examples

Change a field's checkout label in code

```
add_filter( 'alg_wc_ccf_field_label', function ( $label, $data, $field_nr ) {
    if ( 1 === (int) $field_nr ) {
        $label = 'Delivery instructions';
    }
    return $label;
}, 10, 3 );
```

Hide a field for a specific condition

```
add_filter( 'alg_wc_ccf_field_visible', function ( $visible, $field_nr, $frontend ) {
    if ( 2 === (int) $field_nr && is_user_logged_in() ) {
        return false;
    }
    return $visible;
}, 10, 3 );
```

Base a percent fee on the subtotal without shipping

```
add_filter( 'alg_wc_ccf_fee_cart_total', function ( $total, $field_nr ) {
    return WC()->cart->get_subtotal();
}, 10, 2 );
```

Note on the field count filter



CUSTOM CHECKOUT FIELDS FOR WOOCOMMERCE

See every feature and the pricing on the Custom Checkout Fields for WooCommerce plugin page.

View plugin →

`alg_wc_ccf_total_fields` is the single most important filter: it decides how many field tabs and front-end fields exist. In the free version it is fixed at `1`. In the Pro version it returns the value of the **Custom fields number** option, up to 100. Overriding it in code without the matching stored options is not supported for adding fields, but reading it is the correct way to loop over the active fields.

Need a hand?

Our team answers every ticket on the Custom Checkout Fields support forum.

Related articles

- Where Custom Checkout Field Data Is Stored (Order Meta Reference)
- Additional Checkout Fields: How to Reset a Settings Section
- Checkout Field Types Reference
- Additional Checkout Fields: Free vs Pro Versions Compared
- Custom Checkout Fields in Emails and Order Pages: Templates and Placeholders