SL WooCommerce Orders Import

Bulk-import historical or migrated orders with drag-and-drop field mapping

Version 0.2.0 | WordPress Plugin User Manual

Streamline Diving • streamlinediving.com/plugins

Contents

1. Overview
2. Installation
3. Before You Import — Disable WooCommerce Emails
4. Uploading a CSV
5. Drag-and-Drop Field Mapping
6. Import Options & Running the Import
7. Progress & Results
8. CSV Format & Field Reference
    a. Field Reference
    b. Sample CSV
    c. Data Formatting Rules
    d. Valid Values
    e. Tips & Best Practices
9. Product Handling
10. HPOS Compatibility & the Import Block
    a. Using the Import Block (Gutenberg)
11. Troubleshooting & Quick Reference
12. Version History

1. Overview

SL WooCommerce Orders Import bulk-imports historical or migrated orders into WooCommerce from a CSV file, using a visual drag-and-drop field mapping interface — connect your CSV columns to WooCommerce order fields without writing any code.

- **Drag & Drop Mapping** — visually drag CSV columns onto WooCommerce fields
- **CSV Preview** — see the first 5 rows before importing
- **Batch Processing** — imports run in configurable batches to avoid timeouts on large files
- **HPOS Compatible** — built on WooCommerce's modern order API, compatible with High-Performance Order Storage
- **Product Matching** — matches by SKU or name automatically, and creates new products if none is found
- **Non-Intrusive** — only loads on admin pages or pages containing the import block; never interferes with checkout

Component	Minimum
WordPress	5.8
WooCommerce	5.0 (tested up to 8.0)
PHP	7.4

> **WooCommerce required:** the plugin shows an admin notice and will not process imports without an active WooCommerce installation.

2. Installation

1. Go to **Plugins → Add New → Upload Plugin**, select the ZIP, click **Install Now**, then **Activate**.
2. You'll see an **Import Orders** action link on the Plugins page, and an **Orders Import** item in the admin sidebar.

> **Permissions:** you need the `manage_woocommerce` capability — Administrators and Shop Managers have this by default.

3. Before You Import — Disable WooCommerce Emails

> **You must disable WooCommerce emails before every import — this is the first thing the plugin itself warns about, prominently, above the upload area.** If email notifications are enabled, every customer in your import file receives order confirmation and processing emails — even for historical or past orders you're only importing for record-keeping.

> **Do not rely on the "Send email notifications" checkbox (Section 6) to prevent this:** it may not work in all cases. Always manually disable emails in WooCommerce's own settings before every import — treat the checkbox as an extra precaution only, never as your actual safeguard.

1. **Step 1 — Disable emails first:** go to **WooCommerce → Settings → Emails** and turn off the notification emails you don't want sent (New Order, Processing Order, Completed Order, etc.).
2. **Step 2 —** return to the Orders Import page and run your import.
3. **Step 3 —** re-enable your emails in WooCommerce settings once the import is finished, so normal store operations resume sending them.

4. Uploading a CSV

Before You Upload — Two Buttons at the Top of the Page

Above the upload area itself, two buttons are available if you need a starting point:

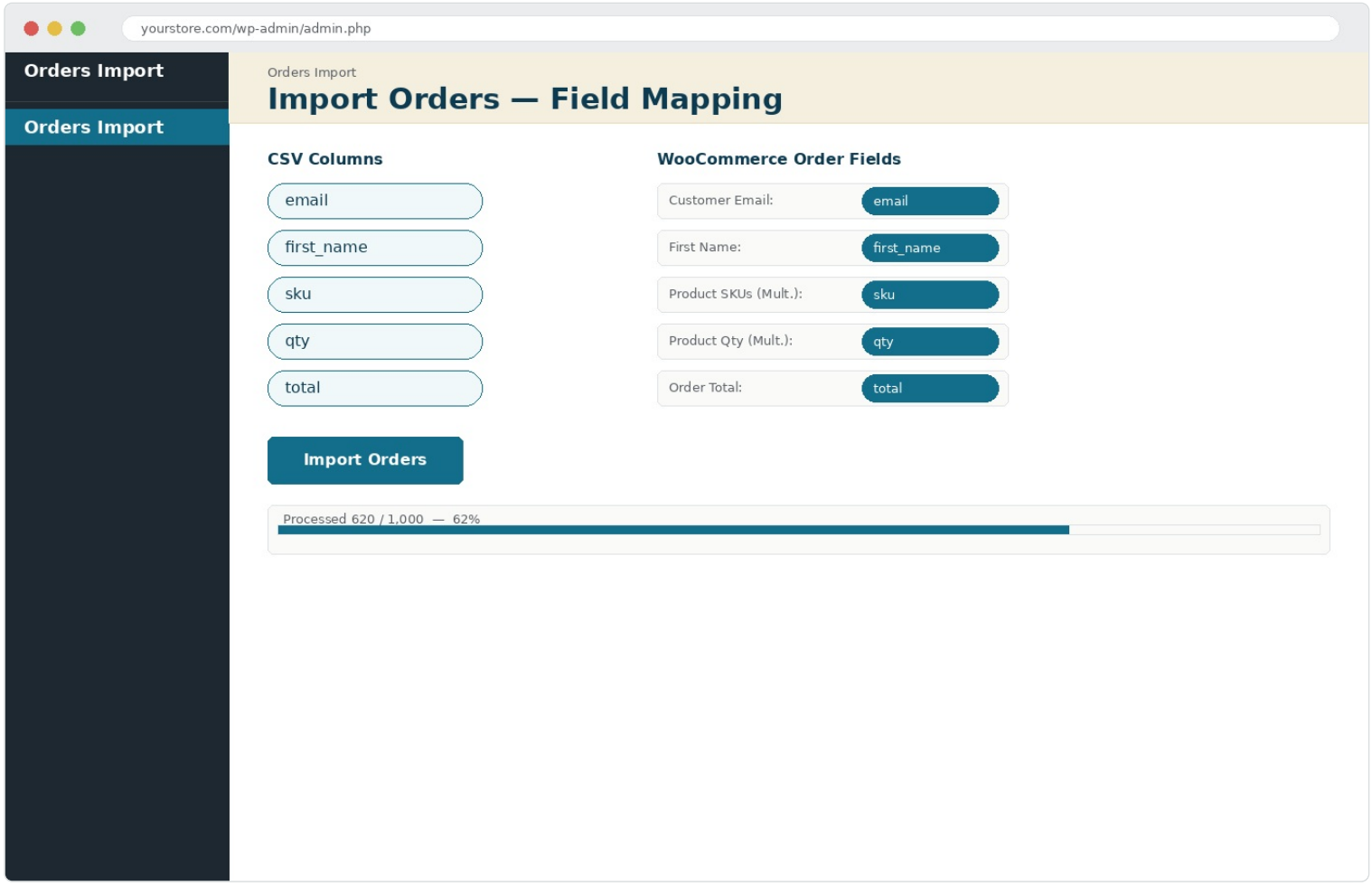
- **⬇ Download CSV Template** — immediately downloads a ready-to-fill CSV file (`streamline-orders-import-template.csv`) with the correct column headers and a few sample rows already in it. Open it in Excel, Google Sheets, or Numbers, replace the sample rows with your real data, and save/export as CSV.
- **Template Guide** — opens a separate, dedicated reference page (in a new tab) with its own full walkthrough: a template preview, the complete field reference broken out by section (Customer Information, Billing Address, Shipping Address, Products, Order Totals, Order Details), data formatting rules, valid values for status/quantity/payment fields, and its own tips list. Section 8 of this manual covers the same ground.

Uploading Your File

Drag and drop your CSV file onto the upload area, or click **Select CSV File** to browse. Only `.csv` files are accepted. Once uploaded, the interface shows the number of columns detected, the total data row count, and a preview of the first 5 rows.

> **Large files:** there's no hard file-size limit in the plugin itself, but your server's `upload_max_filesize` and `post_max_size` settings apply. For 10,000+ row files, use a batch size of 10–25.

5. Drag-and-Drop Field Mapping



Field mapping — drag each CSV column onto its matching WooCommerce order field.

The mapping interface has two panels: the left lists every CSV column header as a draggable chip, and the right lists all importable WooCommerce order fields as drop zones.

1. Find your CSV column in the left panel (e.g. "Email Address").
2. Drag it onto the matching field on the right (e.g. "Customer Email").
3. The chip appears inside the drop zone, confirming the mapping.
4. Repeat for every column you want to import — you don't need to map all of them.

To remove a mapping, drag the chip back to the left panel, or click the × that appears on it while it's in a drop zone.

> **Multi-product columns:** fields marked "(Multiple)" — Product Names, Product SKUs, Product Quantities, and Product Prices — accept several column mappings at once, so orders with more than one line item work correctly.

6. Import Options & Running the Import

Option	Description
Send email notifications	When checked, WooCommerce sends its standard order emails during import. Unchecked by default, for a silent import.
Batch Size	Orders processed per server request: 10, 25 (default), 50, or 100. Use smaller batches for large imports or shared hosting.

> **Keep email notifications off** when importing historical orders — sending emails for old orders will confuse customers.

1. Complete field mapping for every column you want to import.
2. Set your batch size and email notification preference.
3. Click **Import Orders** — active once at least one field is mapped.
4. The import runs immediately, batch by batch.

Click **Cancel Import** to abort a running import. Orders already created remain in WooCommerce — partial imports are not automatically rolled back.

7. Progress & Results

Indicator	Meaning
Progress bar + %	Overall completion
Processed	Total rows read so far
Success	Orders successfully created
Errors	Rows that could not be imported
Total	Total rows in the CSV

If any rows fail, an **Error Details** section expands showing the row number and the specific error for each — commonly invalid email addresses, unreadable dates, or missing required data. Imported orders appear immediately in **WooCommerce → Orders**, where you can verify them, add notes, or update statuses.

8. CSV Format & Field Reference

- File must be standard comma-separated values (`.csv`)
- The first row must be a header row — column names can be anything; you map them visually
- Use UTF-8 encoding to avoid character issues with international names/addresses
- Dates in a standard format: `YYYY-MM-DD`, `MM/DD/YYYY`, or `Month DD, YYYY`
- Monetary values can include symbols and commas (e.g. `$1,234.56`) — stripped automatically
- Empty rows are skipped automatically

Field Reference

Field	Notes
Customer Email	Required for matching existing customers; must be valid
First / Last Name, Phone Number	Billing contact fields
Billing / Shipping Address	Company, Address Line 1/2, City, State, Postal Code, Country (2-letter ISO code recommended) — mapped separately so billing and shipping can use different source columns
Product Names / SKUs (Multiple)	SKU matching is preferred over name matching
Product Quantities (Multiple)	Defaults to 1 if not mapped
Product Prices (Multiple)	Per-unit price; falls back to the product's current WooCommerce price if not mapped
Invoice Number	Saved as order meta (`_invoice_number`)
Order Total	If mapped, sets the total directly; otherwise WooCommerce calculates it
Shipping Cost / Tax Amount	Added as "Imported Shipping" / "Imported Tax" line items
Order Status	pending, processing, on-hold, completed, cancelled, refunded, or failed — defaults to "processing"
Order Date	Defaults to the import date if not mapped
Order Notes	Added as internal order notes visible in WooCommerce admin
Payment Method	Stored as the payment method title (e.g. "Credit Card", "PayPal")

Sample CSV

```
email,first_name,last_name,sku,product_name,qty,price,total,status,date
john@example.com,John,Smith,DIVE-001,Wetsuit 5mm,1,129.99,151.64,completed,2025-06-15
```

> **Multi-product orders:** for orders with several line items, add extra columns (e.g. `sku2`, `product2`, `qty2`, `price2`) and map each set to the corresponding "(Multiple)" field.

Data Formatting Rules

Field	Rule
Dates	`YYYY-MM-DD` is recommended since it's unambiguous, but `MM/DD/YYYY`, `DD-MM-YYYY`, "Month DD, YYYY", and "DD Month YYYY" are all accepted
Currency / Prices	`129.99`, `$129.99`, and `$1,299.99` are all accepted — symbols and commas are stripped automatically. Use `0.00` or `0` for a genuine zero value (free shipping, no discount).
Country & State Codes	Use standard two-letter ISO codes for best compatibility — e.g. US, GB, CA, AU, NZ, DE, FR, JP, MX for country; FL, CA, NY, TX, etc. for US states
Empty Cells	Leave optional cells genuinely blank — don't type "N/A", "null", or "none", since these can be imported as literal text into the order
Special Characters & Commas	If a value contains a comma (e.g. an address like "Suite 4, Building B"), make sure your spreadsheet app wraps it in double quotes when exporting to CSV — all major spreadsheet apps do this automatically, so this is rarely something to fix by hand

Valid Values

Field	Accepted Values
Order Status	pending, processing, on-hold, completed, cancelled, refunded, failed — case-insensitive, with or without the `wc-` prefix. An unrecognized value falls back to processing.
Product Quantities	A whole positive number only (1, 2, 3…) — decimal quantities aren't supported. Defaults to 1 if left blank.
Payment Method	Free text — whatever label makes sense for your store (Credit Card, PayPal, Cash, Bank Transfer, Check, Gift Card, Store Credit, Square, Stripe, etc.)

Tips & Best Practices

- **Test with a small batch first** — import 5–10 rows, verify the orders in WooCommerce, then import the rest
- **Use SKUs over product names** for matching — SKU lookup is exact and fast, while name lookup can miss on any spelling difference
- **Use a batch size of 10–25** on shared hosting to avoid PHP timeout errors
- **Keep one row per order** — a multi-product order uses multiple column sets on a single row (Section 8, Sample CSV above), not multiple rows
- **Don't map Order Total if you have individual line-item prices** — let WooCommerce calculate the total automatically from line items, shipping, and tax, so it stays internally consistent
- **Check auto-created products after import** — anything not matched by SKU or name gets created automatically (Section 9); review those in WooCommerce → Products afterward and fill in descriptions, images, and categories
- **Save a backup of your CSV before importing** — the importer cannot undo orders once they've been created

9. Product Handling

When the importer processes a product line, it looks the product up in this order:

1. **Find by SKU** — if a SKU column is mapped and non-empty
2. **Find by Name** — if no SKU match, searches published products by exact name
3. **Create new product** — if nothing matches, a new simple product is automatically created (published, with the given name, SKU, and price)

> **Auto-created products are published and visible in your store.** Review them after import and fill in descriptions, images, categories, and stock as needed.

10. HPOS Compatibility & the Import Block

The plugin uses WooCommerce's modern order API (`wc_create_order()`, `WC_Order` setters, `$order->save()`) rather than direct database manipulation, so it works correctly with both High-Performance Order Storage (HPOS) and the legacy post-based order storage — it auto-detects which is active.

Using the Import Block (Gutenberg)

1. Open a post or page in the block editor.
2. Click the + inserter and search for "SL WooCommerce Orders Import".
3. Insert the block — the full import UI renders inline.
4. Publish or preview the page.

> **Frontend access is permission-gated:** only users who can `manage_woocommerce` can use the import tool on a frontend page — regular visitors never see the interface.

11. Troubleshooting & Quick Reference

Issue	What to check
"Import Orders" menu not showing	Confirm WooCommerce is active and you have manage_woocommerce capability, then deactivate/reactivate the plugin and hard-refresh.
CSV upload fails	Confirm the file has a .csv extension and is within your server's upload size limits.
Rows fail with date errors	Use a standard format (YYYY-MM-DD, MM/DD/YYYY, or Month DD, YYYY).
Wrong product matched	Map the SKU column rather than relying on name matching — SKU takes priority and is exact.

Start a new import	Orders Import → upload CSV → map fields → Import Orders
Import silently (no customer emails)	Leave "Send email notifications" unchecked
Handle a 10,000+ row file	Set Batch Size to 10–25
Embed the importer on a page	Block editor → insert "SL WooCommerce Orders Import" block

12. Version History

What's changed in each release of SL WooCommerce Orders Import, in plain language — for reference and troubleshooting.

Version	Date	What Changed
0.2.0	—	Version bump with no changelog entry recorded — current behavior, including HPOS compatibility and the frontend Import block (Section 10), is documented throughout this manual.
0.1.0	—	Initial release: drag-and-drop CSV field mapping (Section 5), multi-product-per-order support (Section 9), granular address field mapping (Section 8), email notification controls (Section 6), and real-time mapping validation.

> **About this table:** the plugin's readme.txt changelog only records the initial 0.1.0 release, so the specific changes in 0.2.0 can't be individually verified — every future update gets a dated row here going forward.