

# How to Export All WooCommerce Orders using WP-CLI

<https://orbisius.com/blog/export-all-woocommerce-orders-using-wp-cli/>



Did your WordPress site ever crash because bad plugins or themes? Next time you should use [qSandbox](#).

## Introduction

Effective order management is a cornerstone of success in the e-commerce domain. For businesses utilizing WooCommerce on WordPress, sifting through extensive order data can be a significant challenge. This article introduces a streamlined approach to export all orders, including those with varying statuses like refunded, cancelled, and completed, using WP-CLI – a command-line interface for WordPress.

## Understanding WP-CLI

WP-CLI is an essential tool for WordPress site administrators, offering a range of commands to efficiently manage various aspects of WordPress, including WooCommerce. It's particularly useful for handling large datasets and executing complex operations that would be cumbersome through the regular WooCommerce interface.

## Prerequisites

Before proceeding, ensure you have:

1. A WordPress site with WooCommerce.
2. WP-CLI installed on your server (installation guide). Most hosting companies have this already installed.
3. Command line access to your WordPress server.
4. Necessary permissions for accessing and modifying WooCommerce data.

## Why do you need to export the orders?

In this post we should you how to export the data into .CSV (Excel) file format. It can be imported into Excel or Google Sheets.

That way you can process the file however you want. You could even import the file into another tool that can analyze and see trends.

## Export All WooCommerce Orders (all time)

To export all WooCommerce orders since you had your online store you can run this wp-cli command

```
bash
```

```
wp wc shop_order list \  
--user='yourusername' \  
--status="any" \  
--format=csv > your_wc_export_filename.csv
```

## Export All WooCommerce Orders for a specific period

You may need to export all the orders within a given calendar year. You need to pass --before and --after parameters.

The date must be in so called ISO 8601 format. In ISO 8601, dates and times are formatted to allow for easy sorting and compatibility across different systems and regions. Here's a breakdown of the components:

- **YYYY**: Represents the year as a four-digit number.
- **MM**: Represents the month as a two-digit number.
- **DD**: Represents the day as a two-digit number.
- **T**: The letter 'T' is used as a delimiter to separate the date and time components.
- **00:00:00**: Represents the time in a 24-hour format, including hours, minutes, and seconds.

So, YYYY-MM-DDT00:00:00 would be replaced with an actual dates and time, such as 2023-01-01T00:00:00, which would represent midnight on the 15th of March, 2023. This format is widely used because of its clarity making it suitable for a wide range of applications, including programming, data logging, and communication between different systems.

If you want orders for a given calendar year to be display you'd use  
--after="2023-01-01T00:00:00" and --before="2022-12-31T23:59:59"

```
wp wc shop_order list \  
--user='yourusername' \  
--after="YYYY-MM-DDT00:00:00" \  
--before="YYYY-MM-DDT23:59:59" \  
--status="any" \  
--format=csv > your_wc_export_filename.csv
```

## Export WooCommerce Orders for a WordPress Multisite Store

If you're running WordPress Multisite and have one or more stores set up n it's important to specify the --url parameter because WooCommerce needs to know which database tables to access.

```
--url=https://yourwebsite.com/some-subsite
```

if you have the WordPress store set up on a subdmain.

```
--url=https://some-subsite.yourwebsite.com
```

Customize the command with your website's URL, your username, the desired date range, and the export file's name.

### Command Breakdown

- wp wc shop\_order list: Lists WooCommerce shop orders.
- --url: Your WooCommerce store's URL (optional for regular WordPress installs but required for WP Multisite setups).
- --user: Your username with administrative or shop manager permissions.
- --after & --before: Set the desired date range.
- --status="any": Include all order statuses.
- --format=csv: Export in CSV format.
- > your\_export\_filename.csv: Output the data to a CSV file.

We're using a slash \ at the end of the command, so the command doesn't become one long command. When it becomes too long it's very easy to miss some important parameters.

## Important Security Reminder

After downloading the exported file, it is crucial to delete it from the server. Keeping such files with sensitive data can pose a security risk if accessed by unauthorized parties. Always ensure data protection by promptly removing exported files after use. If hackers manage to get a list of the files on the server they will be able to download it and access client data.

It is not recommended to run this as root.

## Conclusion

Exporting WooCommerce orders via WP-CLI is not only time-efficient but also allows for more sophisticated data handling. This method is particularly beneficial when dealing with large volumes of orders and can be an integral part of an advanced WooCommerce management strategy.

Did your WordPress site ever crash because bad plugins or themes? Next time you should use [qSandbox](#).

Disclaimer: The content in this post is for educational purposes only. Always remember to take a backup before doing any of the suggested steps just to be on the safe side. Referral Note: When you purchase through an referral link (if any) on this page, we may earn a commission.