

Echo Scrape

`echo_scrape.py` is a web scraping script designed to automate the process of logging into <https://login.echo-usa.com>, extracting product details, and saving the information into a CSV file. Additionally, the script has the option to upload the output CSV file to an FTP server once scraping is complete.

Features

- Automates the login process using provided credentials.
- Scrapes product information such as brand, item code, description, retail price, quantity on hand, and more.
- Allows FTP upload of the output file, with options to overwrite existing files or append a timestamp to the filename.
- Supports a headless browser mode for silent operation without opening a window.
- Handles product inventory and price data extraction.

Requirements

Minimum System Requirements

- **Operating System:** Windows 10/11 or modern Linux distributions (Ubuntu 20.04+, Fedora 32+, CentOS 8+, etc)
- **Python Version:** 3.7+
- **chromedriver:** Use the latest version of chromedriver to match installed chrome.

Dependencies

You can install the necessary dependencies via the provided `requirements.txt`. Run the following command to install them:

```
pip install -r requirements.txt
```

The `requirements.txt` includes the following libraries:

- `requests`: For making HTTP requests.
- `beautifulsoup4`: For parsing HTML content.
- `selenium`: For automating web browser interactions.
- `lxml`: For faster HTML/XML parsing (used by BeautifulSoup).
- `pyinstaller`: For creating executable files from the Python script.
- `openpyxl`: For optional exporting to xlsx format.

Chromedriver

You must have Chromedriver installed for Selenium to work.

- Windows 10/11: Use the latest version of `chromedriver.exe`.
- Linux: Use the latest version of `chromedriver`

You can download the appropriate version of `chromedriver` from the official website: [ChromeDriver](#).

Once downloaded, place `chromedriver.exe` or `chromedriver` in the same directory as the script.

Configuration (`config.txt`)

The configuration file (`config.txt`) contains the necessary parameters for the script. Here is a breakdown of the configuration options:

```
{
  "login_url": "https://login.echo-usa.com",
  "username": "your_username",
  "password": "your_password",
  "headless_mode": false,
  "max_rows": "all",
  "input_file": "input.csv",
  "output_file": "output.csv",
  "ftp_host": "None",
  "ftp_port": 21,
  "ftp_username": "ftp_user",
  "ftp_password": "ftp_password",
  "ftp_directory": "/path/to/directory",
  "overwrite_existing": true,
  "rsv_qty": 5
}
```

Parameters:

- `login_url`: The URL for logging into the website.
- `username`: Your username for logging into the website.
- `password`: Your password for logging into the website.
- `headless_mode`: Boolean (true or false), if true, the browser will operate in headless mode (no GUI).
- `max_rows`: Set to "all" for scraping all rows or an integer to limit the number of rows to scrape.
- `input_file`: The filename of the input CSV file containing the products to be scraped.
- `output_file`: The filename where the output file will be saved (default is output.csv, but can be .csv or .xlsx)
- `ftp_host`: The FTP server hostname or IP address. If set to "None", the file is saved locally.
- `ftp_port`: The FTP server port (default is 21).
- `ftp_username`: Your FTP username.
- `ftp_password`: Your FTP password.
- `ftp_directory`: The directory on the FTP server where the file will be uploaded.
- `overwrite_existing`: Boolean (true or false), determines whether to overwrite an existing file on the FTP server.

- **rsv_qty**: The default reserved quantity, set to 5 by default.

Running the Script

To run the script, execute the following command in the terminal:

```
python echo_scrape.py
```

or

```
python3 echo_scrape.py
```

Ensure that:

- The configuration file (**config.txt**) is set up with the correct parameters.
- The chromedriver version is correct for your Operating System (see the requirements above).
- The input CSV file exists, and the FTP server information is correctly configured if you intend to upload the file to an FTP server.

FTP Upload:

After scraping is complete, the script will upload the output CSV file to the specified FTP server. If the **overwrite_existing** flag is set to **false**, a timestamp will be appended to the file name before uploading.

Output Fields

The output file will contain the following fields, each representing a specific piece of product information:

- **MFG**
 - **Description:** The manufacturer of the product. This field provides the name of the company that produces the item.
 - **Example:** **Echo**, **Shindaiwa**
- **Type**
 - *Not currently used* - **blank**
- **Subtype**
 - *Not currently used* - **blank**
 - **Item Code**

- **Description:** A unique identifier for the product. This code is often used to track inventory and can be a part number, SKU, or internal identifier.
- **Example:** ECH 12345, BIL 98765
- **Item Description**
 - **Description:** A brief description of the product, often including features, size, or other distinguishing characteristics.
 - **Example:** 20" BAR & CHAIN COMBO, PISTON KIT, 1-RING
- **Retail**
 - **Description:** The retail price of the product. This is the listed MSRP price before any discounts or promotions are applied.
 - **Example:** 199.99, 349.95
- **Qty on Hand**
 - *Not currently used - 0*
- **Qty on Order**
 - *Not currently used - 0*
- **RSV Qty**
 - **Description:** The reserved quantity, which represents the number of units set aside for specific orders or needs. This is typically used in inventory management to allocate stock.
 - **Example:** 0, 5
- **Cost**
 - **Description:** The cost price of the product, typically the price paid to acquire or produce the item. This can be useful for inventory and financial tracking.
 - **Example:** 120.00, 220.50

PROF

Troubleshooting

Common Issues:

- chromedriver version mismatch: Ensure you are using the correct version of **chromedriver** based on your Operating System version.
- FTP connection errors: Check that the FTP credentials and server path in **config.txt** are correct. Ensure that the FTP server is accessible.

License

This script is provided as-is. Use it at your own risk. The author is not responsible for any damage or issues caused by this script.

Contact Information

For any questions, issues, or feedback regarding this script, please reach out:

- **Author:** Jim Tyranski
- **Email:** jim@tyranski.com

Please ensure to provide detailed information about the issue you're experiencing, including any relevant error messages and the configuration details used when running the script.

Changelog

0.5.0 - Updated OS platforms

- Added cross-platform for executing from Linux
- Removed unsupported Windows 7
- Updated requirements.txt to use latest dependencies

0.4.0 - Fixed authentication

- Updated username/password elements when authorizing login page

0.3.0 - Added output support of xlsx

- Will save in .xlsx format if specified in config.txt output_file
- If ftp_host includes "ftp://", connection will still work
- Added code to exclude text from Retail and Cost columns in output file
- Multiple product results now navigate to first result, then scrapes information
- If product not found, output file now prints 'Product Not Found' in subtype column.

0.2.0 - Minor update for scraper and ftp logic

- Set explicit encoding for read/write of CSV files to UTF-8
- If ftp_host is set blank "" in config.txt, it will ignore trying to connect

0.1.0 - Initial Release