

Supermarket Scraper

This project is a web scraping tool designed to extract product data from various supermarket websites. It supports both basic and advanced scraping modes, allowing users to collect essential and enriched product information.

Supported supermarkets are organized in the `supermarkets/` folder, with each supermarket having its own dedicated scraping scripts.

Follow the setup instructions below to get started.

Setup Instructions

1. Install ChromeDriver

Make sure **Google Chrome** is installed.

◆ macOS

```
brew install chromedriver
```

◆ **Windows** 1. Download the version that corresponds to your Chrome version from: <https://googlechromelabs.github.io/chrome-for-testing/> 2. Extract and add the path to chromedriver.exe to your System Environment Variables > Path.

◆ Linux (Debian/Ubuntu)

```
sudo apt update
```

```
sudo apt install -y unzip xvfb libxi6 libgconf-2-4 libnss3 libxss1 libappindicator1 libindicator7
```

```
wget https://chromedriver.storage.googleapis.com/124.0.6367.91/chromedriver_linux64.zip
```

```
unzip chromedriver_linux64.zip
```

```
sudo mv chromedriver /usr/local/bin/
```

Test installation:

```
chromedriver --version
```

2. Create Virtual Environment

```
python -m venv venv
```

macOS/Linux

```
source venv/bin/activate
```

Windows

```
venv\Scripts\activate
```

3. 📦 Install Dependencies

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

🚀 Running the Scraper

To run a scraper for a specific supermarket, use the run.py script.

1. Navigate to the root directory of the project.
2. Open the `supermarkets/` folder to view the available supermarket directories.
3. Choose one of the folder names (e.g., parma) and replace `{supermarket_name}` in the command below:

```
python run.py {supermarket_name} basic
```

◆ Basic Mode

Runs only `basic_scraper.py` for the selected supermarket, collecting essential product data such as name, price, and other details from the category listing page.

```
python run.py {supermarket_name} basic
```

◆ Advanced Mode

Runs both `basic_scraper.py` and `advanced_scraper.py`. The advanced scraper enriches the data with extra information where available.

```
python run.py {supermarket_name} advanced
```

⚠ Note: The availability and type of advanced data depend on the supermarket's website. Some fields (like brand or origin) may be missing or incomplete for certain supermarkets.

📝 Example Commands

👉 Run Parma's basic scraper.

```
python run.py parma basic
```

👉 Run Yerevan City's full (basic + advanced) scraping pipeline.

```
python run.py yerevan_city advanced
```

Output

Results are saved to `output_data/<supermarket>/` .

Filenames are timestamped for versioning (e.g. `basic_31.05.25_13.45.22.xlsx`).

Notes

`basic_scraper.py` : Scrapes product data like name, price, and URL.

`advanced_scraper.py` : Enriches the data with brand, origin country, and availability.

Each supermarket has its dedicated scripts under `supermarkets/<name>/` .

Troubleshooting

ChromeDriver not found: Ensure it's installed and added to your PATH.

Version mismatch: Your ChromeDriver version must match your Chrome browser version.

Permission denied (Linux): Run `chmod +x chromedriver` or use `sudo`.