

# E-Commerce Price Tracker

## Introduction

The **E-Commerce Price Tracker** is a Python-based project designed to monitor price changes for products on major e-commerce platforms such as Amazon, eBay, Walmart, and AliExpress. The script fetches product prices at scheduled intervals, stores them in a database, and notifies the user via email when a price drop is detected.

## Features

- Web Scraping: Fetches prices from multiple e-commerce websites.
- Data Storage: Saves the latest price data in an SQLite database.
- Price Comparison: Compares the new price with the previously recorded price.
- Email Notifications: Sends alerts when a price drop is detected.
- Configurable Settings: Product URLs, email credentials, and user-agent headers are stored in a configuration file.

## Requirements

### Software & Libraries

Python 3.x (Ensure Python is installed)

Required Python Libraries:

- requests (for fetching webpage data)
- BeautifulSoup4 (for HTML parsing)
- sqlite3 (for database storage)
- smtplib (for sending email notifications)
- email.mime (for formatting emails)

To install dependencies, run:

```
pip install requests BeautifulSoup4
```

### Account Credentials

SMTP Email Account: Required to send notification emails.

Valid Product URLs: Ensure the URLs used for tracking are correct.

## Project Structure

```
E-Commerce-Price-Tracker/  
  
|-- main.py      # Entry point of the script  
  
|-- scraper.py   # Web scraping logic  
  
|-- database.py  # Database handling  
  
|-- notifier.py  # Email alert system  
  
|-- config.py    # Configuration settings (URLs, headers, SMTP details)  
  
|-- prices.db    # SQLite database storing product prices
```

## How It Works

- Scraping Prices: Extracts product prices from predefined e-commerce websites using BeautifulSoup.
- Storing Data: Prices are stored in an SQLite database (`prices.db`). If a product exists, it updates the price.
- Price Drop Detection: Compares the latest price with the previous price and sends an email if a drop is detected.
- Sending Notifications: Uses SMTP to send an email alert containing product name, old price, new price, and purchase link.

## Usage Instructions

### 1. Setup Configuration

Edit `config.py` and update the following details:

```
PRODUCTS = {  
    "Amazon": "<Amazon Product URL>",  
    "eBay": "<eBay Product URL>",  
    "Walmart": "<Walmart Product URL>",  
    "AliExpress": "<AliExpress Product URL>"  
}  
  
SMTP_SERVER = "smtp.example.com"  
SMTP_PORT = 587  
SENDER_EMAIL = "your_email@example.com"  
RECEIVER_EMAIL = "receiver@example.com"  
EMAIL_PASSWORD = "your_password"
```

## 2. Run the Script

Execute the script from the terminal:

```
python main.py
```

## 3. Check the Output

If a price drop is detected, you will receive an email alert.

The script logs price-fetching status and errors in the console.

## Example Output

### **\*\*Console Output:\*\***

```
Fetching data from Amazon...
Fetching data from eBay...
Fetching data from Walmart...
Fetching data from AliExpress...
Price drop detected for Amazon: $499.99 -> $459.99
Alert sent for Amazon!
Price tracking completed.
```

### **\*\*Email Alert:\*\***

Subject: Price Drop Alert for Amazon Product!

The price for Amazon Product has dropped from \$499.99 to \$459.99!  
Check it here: <https://www.amazon.com/dp/B08N5WRWNW>

## Future Enhancements

- Support for additional e-commerce platforms.
- Implement a GUI interface for easier tracking.
- Add scheduling for automatic price tracking.
- Integrate SMS notifications.

## Conclusion

This project efficiently tracks price changes and alerts users of significant price drops. With modular components for scraping, database management, and email notifications, it serves as a valuable tool for e-commerce enthusiasts looking to grab the best deals.