

## Price Comparison Tool

### Detailed Project Description:

Aim to build a tool that can compare prices from different websites and display optimal prices for a given product in one place.

To achieve this goal, we focus on the following:

1. Fetching prices from two or more different websites. (Web Scraping)
2. Storing data in the CSV file.
3. Displaying the lowest price website
4. Visualize prices
5. Send-lowest price website to the user input email

**Web Scraping:** Web scraping is a process of collecting relevant information from a particular webpage and then exporting that information in a proper format according to our needs

**Git hub Link:** [git@github.com:Lakshmi1536/Price-Comparison-Tool.git](https://github.com/Lakshmi1536/Price-Comparison-Tool.git)

### Programming Concepts used in this Project

#### Libraries/Packages used in this project:

- BeautifulSoup-Helps in extracting data out of markup languages like HTML and XML
- requests- “pip install requests”- command to install the requests library. It provides a simple and elegant API for interacting with web services and consuming data from various sources
- smtplib- module provides a convenient way to interact with SMTP (Simple Mail Transfer Protocol) servers for sending email. This program used Outlook SMTP server.
- Matplotlib- For creating static, animated, and interactive visualization
- tkinter- GUI toolkit
- csv- comma-separated value files

To Populate headers, we use “User Agent”. To get a user agent google “my user agent”.

### Techniques Learned in this project:

Web Scraping, Data Visualization, methods, GUI tool creation, and email generated using smtplib.

## Screenshots:

Price Comparison Tool

Select your product and get the best deal

Products

Apple iPad

Apple Watch

Fitbit Watch

Pixel Phone

Ecommerce Websites

Amazon

EBAY

Walmart

Email

Email Notification

Please enter your email:

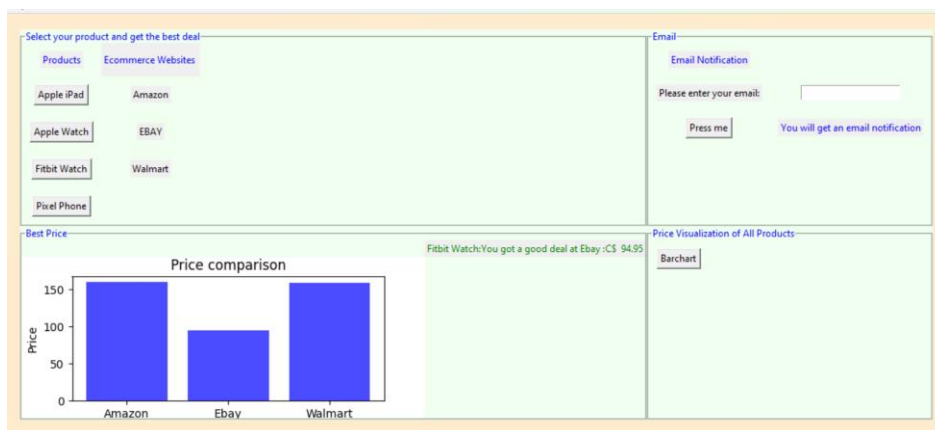
Press me

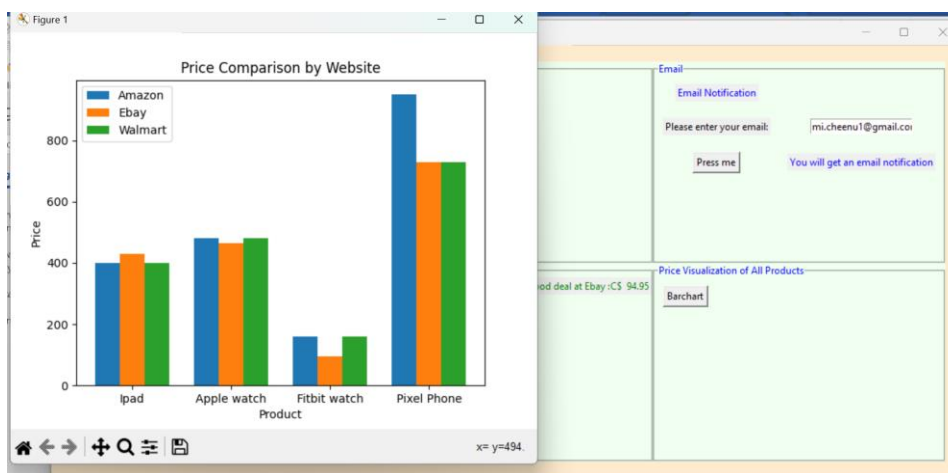
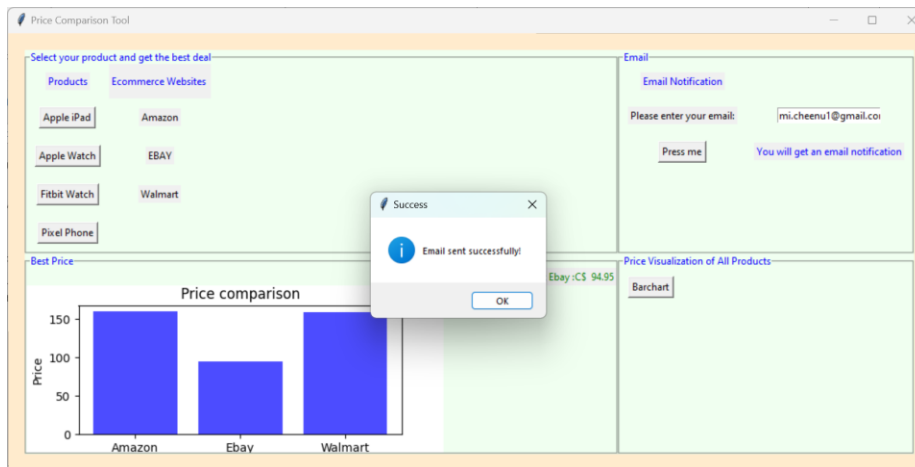
You will get an email notification

Best Price

Price Visualization of All Products

Barchart





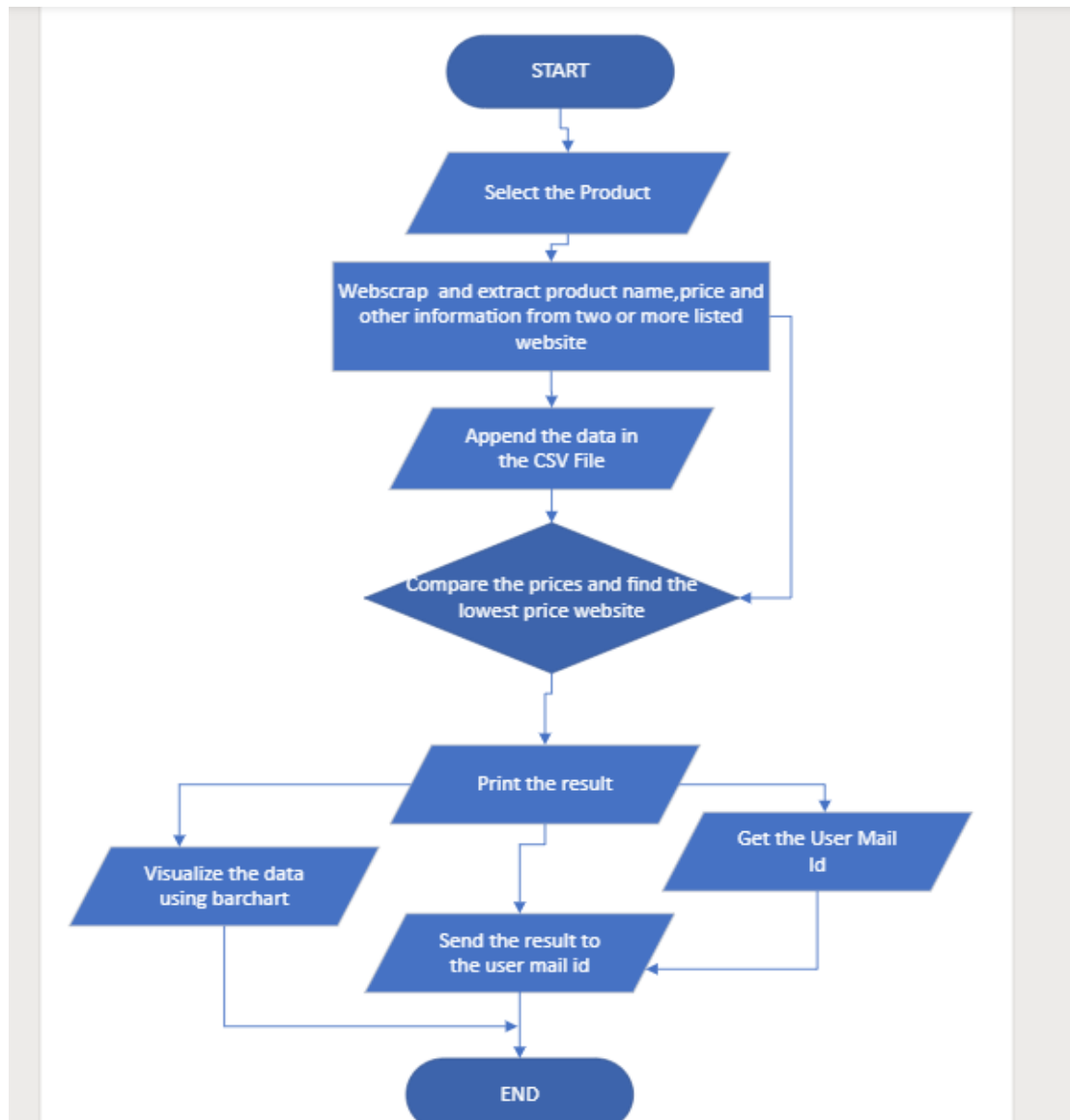
CSV File:

```

1 Website_Name,Product_Name,Price,URL
2
3 Amazon,"Apple iPad (9th Generation): with A13 Bionic chip, 10.2-inch Retina Display, 64GB",429.99,https://www.amazon.com/dp/B08FMN4C4R
4
5 ebay,Apple iPad 9th Generation 10.2-inch 64GB Wi-Fi Space Gray (2021 Model) A2602 New,429.99,https://www.ebay.com/itm/1234567890
6
7 Amazon,"Apple Watch Series 9 [GPS + Cellular 41mm] Smartwatch with Midnight Aluminium Case",429.99,https://www.amazon.com/dp/B08FMN4C4R
8
9 ebay,Apple Watch Series 9 41mm GPS With Sport Band - NEW SEALED,464.55,https://www.ebay.com/itm/1234567890
10

```

Overall Flow of the program:



### Functions:

**def best\_ipad():** Compare the price by web scraping two or more websites and display the lowest price website detail.

**def best\_apple\_watch():** Compare the price by web scraping two or more websites and display the lowest price website detail.

**def show\_bmi\_barchart():** display the bar chart by comparing prices and websites from the functions like best\_ipad,best\_apple\_watch etc.,

**def send\_email():** Generate an email with the result "Lowest price website" to the user input email.