**Implementing Web Scrapping in Python with**

**Beautiful Soup**

As a project work for Course

PYTHON PROGRAMMING (INT 213)

Name : Dandamudi Priya Sivani

Registration number : 12015705

Name : Ritika Saha

Registration number : 12015741

Name : Mahesh kumar Sai

Registration number : 12015747

Program : CSE B.TECH

Semester : Third

School : School of Computer Science and Engineering

Name of University : Lovely professional University

Date of Submission : 03-12-2021



*Implementing Web Scrapping in Python with Beautiful Soup*

*Abstract :*

Web Scrapping is an automatic way to obtain large amounts of data from the websites . Most of the data is unstructured data in an HTML format which is then converted into structured data . We have implemented this in Python using GUI concept ( Graphical User Interface )

You will only need basic knowledge about the python , request module and some intermediate knowledge about the Tkinter library as well.

*Acknowledgement :*

We would like to thank our mentor – Prof.Ishan kumar for his advice and inputs on this project .Many thanks to my friends and seniors as well , who spent countless hours to listen and provide feedbacks .

*Table of contents*

|  |  |
| --- | --- |
| 1. Abstract |  |
| 1. Introduction    1. Context    2. Motivation    3. Idea |  |
| 3. Team members with roles  3.1 Members   * 1. Contribution |  |
| 4. Modules |  |
| 5. Conclusion |  |

INTRODUCTION :

*2.1 Context :*

This project has been done as part of my course for the CSE at Lovely Professional University. Supervised by Ishaan Kumar , I have one month to fulfill the requirements in order to succeed the module.

*2.2 Motivation :*

The group project was a great occasion to give us the time to learn and confirm our interest for this field. The fact that we can make estimations, predictions and give the ability for Python to learn by themselves is both powerful and limitless in term of application possibilities.

*2.3 Idea :*

As a first experience, we wanted to make my project as much didactic as possible by approaching every different steps of the Tkinter process and trying to understand them deeply. The goal was to implement Web Scrapping with Beautiful Soup

TEAM MEMBERS :

* Ritika Saha

Contributions :

1. Coding (joined)
2. GUI part (joined)
3. PPT

* Dandamudi Priya Sivani

Contributions :

1. Coding (joined )
2. GUI part (joined )
3. Report (joined)

* Mahesh Kumar Sai

Contributions :

1. Coding (joined)
2. Gui part (joined )
3. Report (joined)

*MODULES:*

Tkinter :

Tkinter is a Python tool for creating graphical user interfaces (GUIs). Because it is simple and straightforward to use, it is one of the most often used Python modules for constructing GUI applications. The Tkinter module is already included with Python, so you don't have to bother about installing it individually. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit. Creating a GUI application using Tkinter is an easy task.

All you need to do is perform the following steps :

* Import the Tkinter module.
* Create the GUI application main window.

Requests :

The requests module allows you to send HTTP requests using Python . The HTTP request returns a Response Object with all the response data (content, encoding, status, etc).

CSV :

CSV (Comma Separated Values) is a simple file format used to store tabular data, such as a spreadsheet or database. A CSV file stores tabular data (numbers and text) in plain text. Each line of the file is a data record

Bs4 – Beautiful Soup 4 :

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work

Lxml :

lxml is a Python library which allows for easy handling of XML and HTML files, and can also be used for web scraping. There are a lot of off-the-shelf XML parsers out there, but for better results, developers sometimes prefer to write their own XML and HTML parsers

Resizable :

The resizable() function is used to allow the Tkinter root window to modify its size according to the user's needs, as well as to prevent the Tkinter window from being resized.

Geometry :

Python has capability to create GUI applications using the Tkinter library. The library provides many methods useful for GUI applications. The geometry method is a fundamental one which decides the size, position and some other attributes of the screen layout we are going to create.

Title :

The title() method in Python is a Python String Method that converts the initial letter in each word to Uppercase and the remaining characters in the string to Lowercase before returning a new string.

Widgets of tkinter :

Widgets are GUI components in Tkinter that allow users to interact with the programme using different controls such as Labels,Buttons, Combo Boxes, Check Boxes, Menu Bars, Radio Buttons, and so on.

MessageBox :

MessageBox Widget is used to display the message boxes in the python applications. This module is used to display a message using provides a number of functions.

Label :

Tkinter Widget class, which is used to display text or an image. The label is a widget that the user just views but not interact with .

root :

root is the root window into which all other widgets go. It is an instance of the class Tk, and every tkinter application must have exactly one instance of this class .

Button :

The Button widget is used to add buttons in a Python application. These buttons can display text or images that convey the purpose of the buttons. You can attach a function or a method to a button which is called automatically when you click the button.

Entry :

The Entry widget is used to provide the single line text-box to the user to accept a value from the user. We can use the Entry widget to accept the text strings from the user

Frame :

It works like a container, which is responsible for arranging the position of other widgets.

Text ;

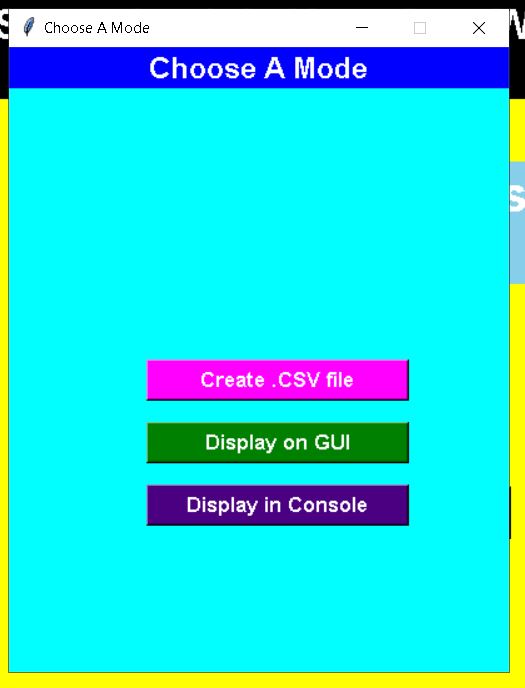
It is used when an user wants to enter multiple text line fields.

SCREENSHOTS

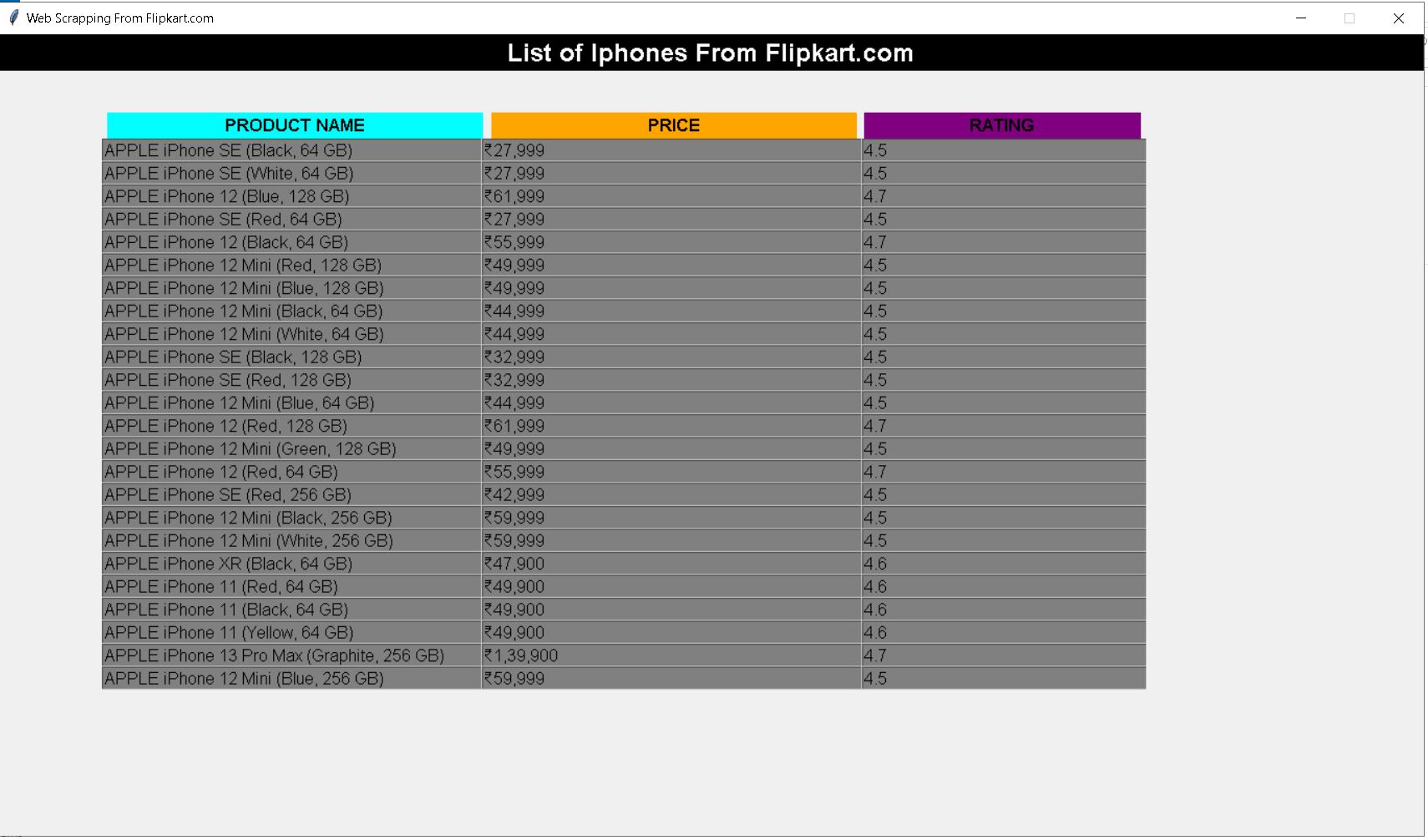
Main Page:-



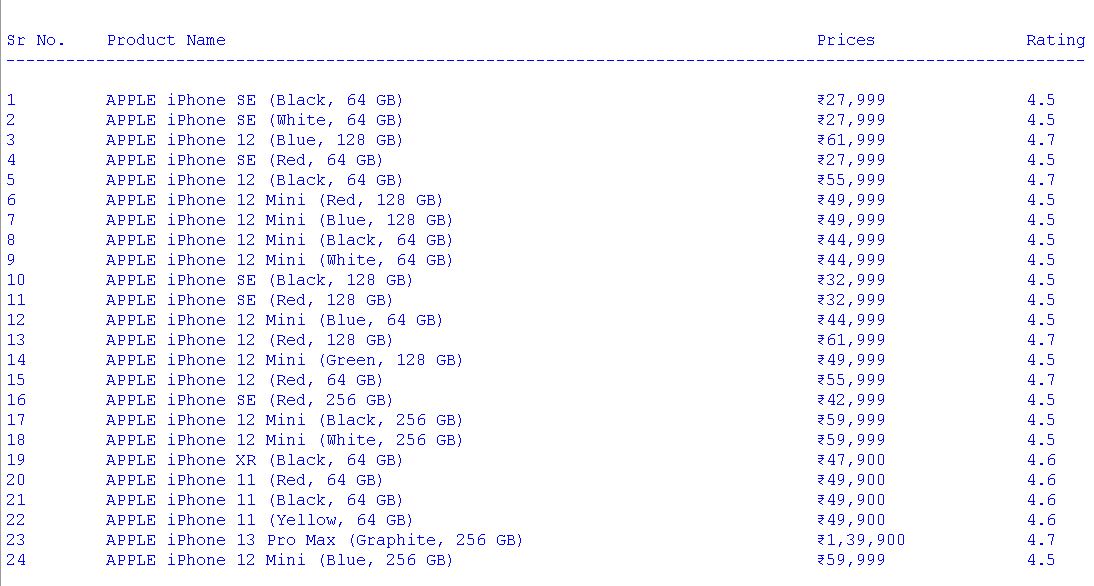
Choosing A Module:-



Display On GUI:-



Display In Console:-

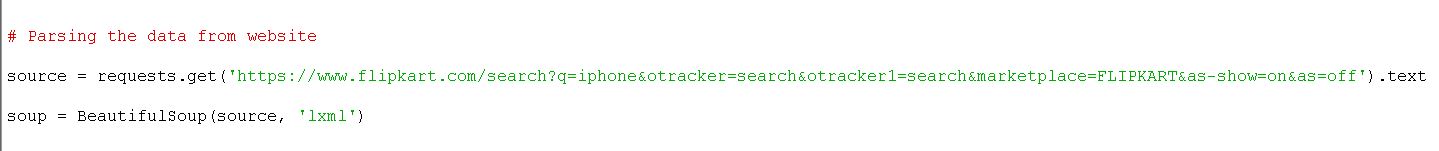


CODE EXPLAINED IN BRIEF STEPS :

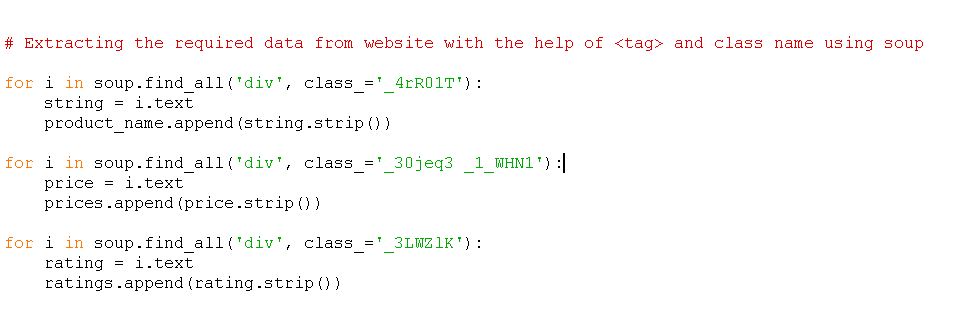
Step 1 :- Installing the required libraries



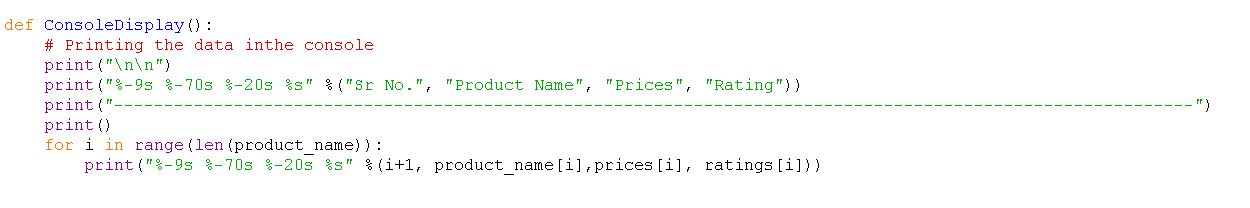
Step 2:- Accessing the HTML content from webpage



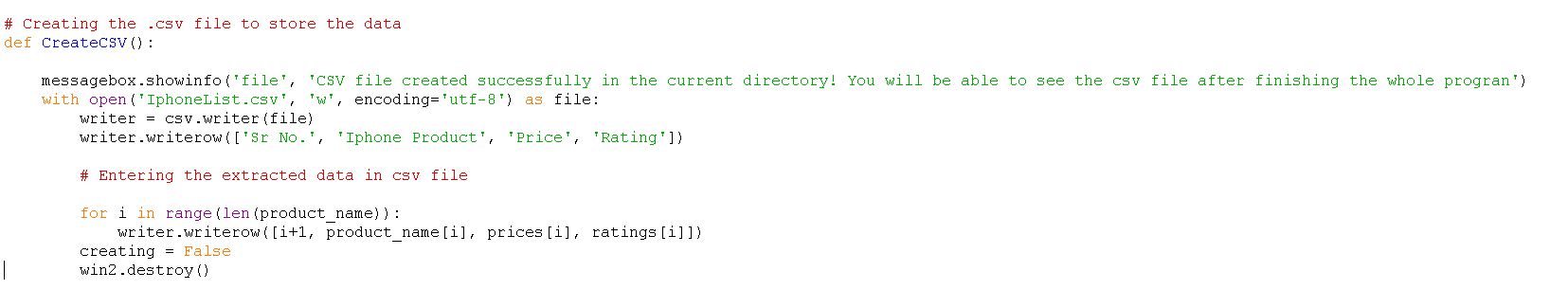
Step 3:- Parsing the HTML content



Step 4:- Printing Data in the Console

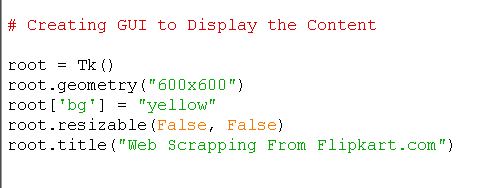


Step 5:- We would like to save all our data in some CSV file

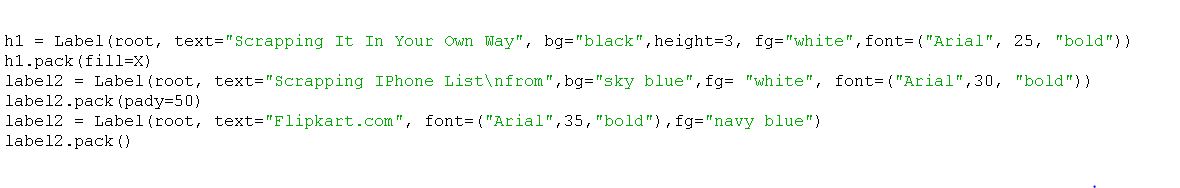


Step 6:- Creating GUI And Tkinter Modules Used:

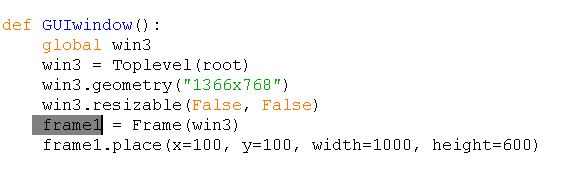
1. Root
2. Geometry



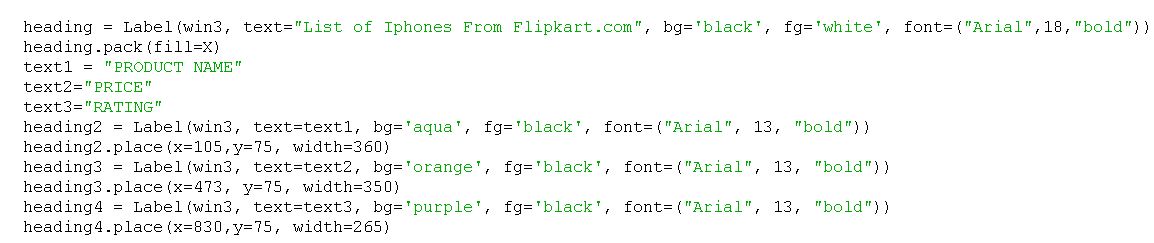
3)Label



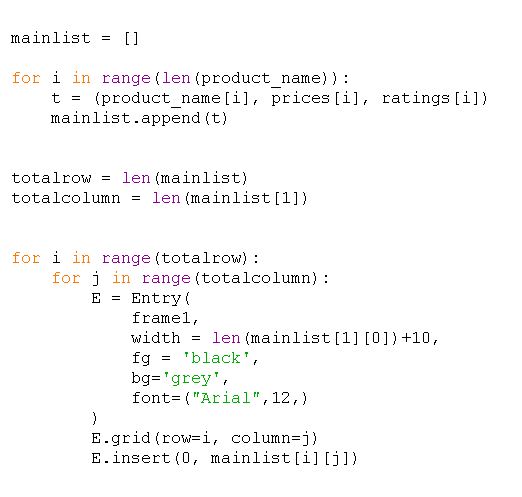
4)Frame



5)Text



6)Entry



7)Button



Conclusions:-

It is our team’s hope that this report will be of immense help for

understanding of our small project.

It was a wonderful and learning experience for us while working on this project.

The project took us through various phases of Web Scraping with Python using beautiful soup. We tried our best to include all the necessary points that are required related to the given topic. Some of the things we did in the project were taken from the internet.

We do hope that our project was interesting and also knowledgeable.