

**Name:**Harsha KG

**Reg\_No:**2248035

**University:**Christ(Deemed to be)University, Bangalore

**Email:** [harshakgkallangot@gmail.com](mailto:harshakgkallangot@gmail.com)

<mailto:harshakgkallangot@gmail.com>), [harsha.kg@msds.christuniversity.in](mailto:harsha.kg@msds.christuniversity.in)

<mailto:harsha.kg@msds.christuniversity.in>)

**Mb:** +917593974960

## Problem Statement

write a function in Python to scrape tabular information from the Agmarket website

## Approach

I got commodity as "Ashgourd" for filtering the data

[https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx\\_Commodity=83&Tx\\_State=0&Tx\\_District=0&Tx\\_Market=0&DateFrom=24-Nov-2021&DateTo=24-Nov-2021&Fr\\_Date=24-Nov-2021&To\\_Date=24-Nov-2021&Tx\\_Trend=0&Tx\\_CommodityHead=Ashgourd&Tx\\_StateHead=--Select--&Tx\\_DistrictHead=--Select--&Tx\\_MarketHead=--Select--](https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=83&Tx_State=0&Tx_District=0&Tx_Market=0&DateFrom=24-Nov-2021&DateTo=24-Nov-2021&Fr_Date=24-Nov-2021&To_Date=24-Nov-2021&Tx_Trend=0&Tx_CommodityHead=Ashgourd&Tx_StateHead=--Select--&Tx_DistrictHead=--Select--&Tx_MarketHead=--Select--)  
([https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx\\_Commodity=83&Tx\\_State=0&Tx\\_District=0&Tx\\_Market=0&DateFrom=24-Nov-2021&DateTo=24-Nov-2021&Fr\\_Date=24-Nov-2021&To\\_Date=24-Nov-2021&Tx\\_Trend=0&Tx\\_CommodityHead=Ashgourd&Tx\\_StateHead=--Select--&Tx\\_DistrictHead=--Select--&Tx\\_MarketHead=--Select--](https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=83&Tx_State=0&Tx_District=0&Tx_Market=0&DateFrom=24-Nov-2021&DateTo=24-Nov-2021&Fr_Date=24-Nov-2021&To_Date=24-Nov-2021&Tx_Trend=0&Tx_CommodityHead=Ashgourd&Tx_StateHead=--Select--&Tx_DistrictHead=--Select--&Tx_MarketHead=--Select--))

1.Import Packages

2.Function to Extract Table info from the website (<https://agmarknet.gov.in>  
(<https://agmarknet.gov.in>))

3.List of Urls to scrape based on the condition given for Ashgourd

4.Create a dataframe and store the values scraped from website

5.Save the data to CSV file

## Import Packages

```
In [1]: ▶ import requests
        from bs4 import BeautifulSoup
        import pandas as pd
```

## Created a Function (extract\_data) to Extract Table info from the website (<https://agmarknet.gov.in>)

```
In [2]: ▶ def extract_data(url):
        # Send GET request to the URL
        response = requests.get(url)

        # Parse the HTML content using BeautifulSoup
        soup = BeautifulSoup(response.content, 'html.parser')

        # Extract the table from the HTML content
        table = soup.find('table')

        # Convert the table to a pandas dataframe
        try:
            df = pd.read_html(str(table))[0]
        except ValueError:
            df = pd.DataFrame()

        return df
```

#Defines a function called extract-data that takes argument url and extract data from html page using the specified url and retrun data in dataframe

#using request module send the request to extract http request

#Using BeautifulSoup library to parse the html content and create a soup object to search

#find method to locate the first table tag in website from the html content (present in view page source code) and assign to tagrget variable

#use try and except method read html function to convert the target varaible to df

#If the read\_html function encounters an error (i.e. it cannot convert the table variable to a dataframe), an empty dataframe is returned instead.The resulting dataframe is returned as the output of the extract\_data function.

## List of Urls to scrape based on the condition



into a single dataframe.

#A for loop is used to iterate over each url in the urls list.

#For each url, the extract\_data function is called to retrieve the data from the corresponding HTML table on the web page.

#If the resulting dataframe is empty, a message is printed indicating that no data was found for the corresponding URL.

#If the resulting dataframe is not empty, a new column called SI\_no. is added to the dataframe using the insert method. The SI\_no. value for each row is set to the value of the SI\_no variable, which is incremented by 1 for each dataframe.

#The resulting dataframe is appended to the dfs list.

#After all dataframes have been processed, the concat function is used to concatenate all the dataframes in dfs into a single dataframe called data.

## Save the data to CSV file

```
In [5]: # Save the data to a CSV file
data.to_csv('data.csv', index=False)

# Print the final dataframe
data
```

Out[5]:

|   | SI_no. | SI no.        | State Name    | District Name | Market Name   | Group         | Commodity     | Variety       | Grade         | Min Price (Rs/Quintal) |
|---|--------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|------------------------|
| 0 | 1      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 2      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 3      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 4      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 5      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 6      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 7      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 8      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 9      | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 10     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 11     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 12     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 13     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 14     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |

|   | SI_no. | SI no.        | State Name    | District Name | Market Name   | Group         | Commodity     | Variety       | Grade         | Min Price (Rs/Quintal) |
|---|--------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|------------------------|
| 0 | 15     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 16     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 17     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 18     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 19     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |
| 0 | 20     | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found | No Data Found          |

```
In [6]: ▶ # Save the data to a CSV file
#re
```

**Same I did for Bringal also as the one given in the word doc**

**Created a Function (extract\_data) to Extract Table info from the website**  
<https://agmarknet.gov.in>  
<https://agmarknet.gov.in>)

```
In [7]: ▶ def extract_data(url):
# Send GET request to the URL
response = requests.get(url)

# Parse the HTML content using BeautifulSoup
soup = BeautifulSoup(response.content, 'html.parser')

# Extract the table from the HTML content
table = soup.find('table')

# Convert the table to a pandas dataframe
try:
    df = pd.read_html(str(table))[0]
except ValueError:
    df = pd.DataFrame()

return df
```

## List of Urls to scrape based on the condition given for Bringal

```
In [8]: ▶ # Define the list of URLs to scrape
urls = [
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=CG&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=CG&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=CG&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=G0&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
'https://agmarknet.gov.in/SearchCmmMkt.aspx?Tx_Commodity=35&Tx_State=GJ&Tx_
]
]
```

## Create a dataframe and store the values scraped from website

```
In [9]: ▶ # Create an empty list to store dataframes
dfs1 = []

SI_no = 1 # initialize SI no to 1
for url in urls:
    df = extract_data(url)
    if df.empty:
        print("No data found for URL:", url)
    else:
        df.insert(0, "SI No.", SI_no) # add SI no as the first column
        SI_no += 1 # increment SI no by 1
        dfs1.append(df)

# Concatenate all the dataframes into one
result = pd.concat(dfs1)
```

```
In [10]: ▶ # Remove column name 'SI_no'
result=result.drop(['SI no.'], axis=1)
```



```
In [11]: result
```

Out[11]:

|   | SI No. | District Name | Market Name             | Commodity | Variety    | Grade | Min Pric (Rs./Quinta) |
|---|--------|---------------|-------------------------|-----------|------------|-------|-----------------------|
| 0 | 1      | Durg          | Durg                    | Brinjal   | Round/Long | FAQ   | 140                   |
| 0 | 2      | Rajnandgaon   | Rajnandgaon             | Brinjal   | Brinjal    | FAQ   | 60                    |
| 0 | 3      | Bilaspur      | Tiphra                  | Brinjal   | Brinjal    | FAQ   | 130                   |
| 0 | 4      | North Goa     | Mapusa                  | Brinjal   | Brinjal    | FAQ   | 280                   |
| 0 | 5      | Ahmedabad     | Ahmedabad               | Brinjal   | Other      | FAQ   | 30                    |
| 0 | 6      | Anand         | Anand(Veg,Yard,Anand)   | Brinjal   | Brinjal    | FAQ   | 100                   |
| 0 | 7      | Bharuch       | Ankleshwar              | Brinjal   | Brinjal    | FAQ   | 200                   |
| 0 | 8      | Navsari       | Bilimora                | Brinjal   | Other      | FAQ   | 120                   |
| 0 | 9      | Amreli        | Damnagar                | Brinjal   | Other      | FAQ   | 130                   |
| 0 | 10     | Banaskanth    | Deesa(Deesa Veg Yard)   | Brinjal   | Other      | FAQ   | 50                    |
| 0 | 11     | Gandhinagar   | Kalol(Veg,Market,Kalol) | Brinjal   | Brinjal    | FAQ   | 100                   |
| 0 | 12     | Kheda         | Kapadvanj               | Brinjal   | Other      | FAQ   | 80                    |
| 0 | 13     | Surat         | Mandvi                  | Brinjal   | Other      | FAQ   | 150                   |
| 0 | 14     | Gandhinagar   | Mansa(Manas Veg Yard)   | Brinjal   | Round/Long | FAQ   | 150                   |
| 1 | 14     | Gandhinagar   | Mansa(Manas Veg Yard)   | Brinjal   | Brinjal    | FAQ   | 150                   |
| 0 | 15     | Rajkot        | Morbi                   | Brinjal   | Brinjal    | FAQ   | 20                    |

|   | SI<br>No. | District Name    | Market Name        | Commodity | Variety | Grade | Min Pric<br>(Rs./Quinta |
|---|-----------|------------------|--------------------|-----------|---------|-------|-------------------------|
| 0 | 16        | Vadodara(Baroda) | Padra              | Brinjal   | Other   | FAQ   | 30                      |
| 0 | 17        | Rajkot           | Rajkot(Ghee Peeth) | Brinjal   | Brinjal | FAQ   | 40                      |
| 0 | 18        | Surat            | Surat              | Brinjal   | Other   | FAQ   | 60                      |
| 0 | 19        | Surendranagar    | Vadhvan            | Brinjal   | Brinjal | FAQ   | 50                      |

**Save the data to CSV file**

```
In [13]: # Save the data to a CSV file
result.to_csv('result.csv', index=False)

# Print the final dataframe
result
```

Out[13]:

|   | SI<br>No. | District Name | Market Name              | Commodity | Variety    | Grade | Min Pric<br>(Rs./Quinta) |
|---|-----------|---------------|--------------------------|-----------|------------|-------|--------------------------|
| 0 | 1         | Durg          | Durg                     | Brinjal   | Round/Long | FAQ   | 140                      |
| 0 | 2         | Rajnandgaon   | Rajnandgaon              | Brinjal   | Brinjal    | FAQ   | 60                       |
| 0 | 3         | Bilaspur      | Tiphra                   | Brinjal   | Brinjal    | FAQ   | 130                      |
| 0 | 4         | North Goa     | Mapusa                   | Brinjal   | Brinjal    | FAQ   | 280                      |
| 0 | 5         | Ahmedabad     | Ahmedabad                | Brinjal   | Other      | FAQ   | 30                       |
| 0 | 6         | Anand         | Anand(Veg,Yard,Anand)    | Brinjal   | Brinjal    | FAQ   | 100                      |
| 0 | 7         | Bharuch       | Ankleshwar               | Brinjal   | Brinjal    | FAQ   | 200                      |
| 0 | 8         | Navsari       | Bilimora                 | Brinjal   | Other      | FAQ   | 120                      |
| 0 | 9         | Amreli        | Damnagar                 | Brinjal   | Other      | FAQ   | 130                      |
| 0 | 10        | Banaskanth    | Deesa(Deesa Veg Yard)    | Brinjal   | Other      | FAQ   | 50                       |
| 0 | 11        | Gandhinagar   | Kalol(Veg,Market,Kalol)  | Brinjal   | Brinjal    | FAQ   | 100                      |
| 0 | 12        | Kheda         | Kapadvanj                | Brinjal   | Other      | FAQ   | 80                       |
| 0 | 13        | Surat         | Mandvi                   | Brinjal   | Other      | FAQ   | 150                      |
| 0 | 14        | Gandhinagar   | Mansa(Manas Veg<br>Yard) | Brinjal   | Round/Long | FAQ   | 150                      |

|   | SI<br>No. | District Name    | Market Name              | Commodity | Variety | Grade | Min Pric<br>(Rs./Quinta) |
|---|-----------|------------------|--------------------------|-----------|---------|-------|--------------------------|
| 1 | 14        | Gandhinagar      | Mansa(Manas Veg<br>Yard) | Brinjal   | Brinjal | FAQ   | 150                      |
| 0 | 15        | Rajkot           | Morbi                    | Brinjal   | Brinjal | FAQ   | 20                       |
| 0 | 16        | Vadodara(Baroda) | Padra                    | Brinjal   | Other   | FAQ   | 30                       |
| 0 | 17        | Rajkot           | Rajkot(Ghee Peeth)       | Brinjal   | Brinjal | FAQ   | 40                       |
| 0 | 18        | Surat            | Surat                    | Brinjal   | Other   | FAQ   | 60                       |
| 0 | 19        | Surendranagar    | Vadhvan                  | Brinjal   | Brinjal | FAQ   | 50                       |

In [ ]:

▶