Activity 5.01: Reading Tabular Data from a Web Page and Creating DataFrames

In this activity, you have been given a Wikipedia page where you have the GDP of all countries listed. You have to create three DataFrames from the three sources mentioned on the page (<https://en.wikipedia.org/wiki/List_of_countries_by_GDP_(nominal)>).

You will have to do the following:

1. Open the page in a separate Chrome/Firefox tab and use something like an Inspect Element tool to view the source HTML and understand its structure.

2. Read the page using bs4.

3. Find the table structure you will need to deal with (how many tables are there?).

4. Find the right table using bs4.

5. Separate the source names and their corresponding data.

6. Get the source names from the list of sources you have created.

7. Separate the header and data from the data that you separated before for the first source only, and then create a DataFrame using that.

8. Repeat the last task for the other two data sources.

Activity 6.01: Handling Outliers and Missing Data

In this activity, we will identify and get rid of outliers. Here, we have a CSV file. The goal here is to clean the data by using the knowledge that we have learned about so far and come up with a nicely formatted DataFrame. Identify the type of outliers and their effect on the data and clean the messy data.

The dataset that we have used here can be found in the visit\_data.csv file. This file contains data generated by a random data generator, and it contains people's names, their gender, email\_id, ip\_address, and the number of visits they made to a particular web page.

The steps that will help you solve this activity are as follows:

1. Read the visit\_data.csv file.

2. Check for duplicates.

3. Check whether any essential column contains NaN.

4. Get rid of the outliers.

5. Report the size difference.

6. Create a box plot to check for outliers.

7. Get rid of any outliers.

The final output should look like this:

After getting rid of outliers the new size of the data is - 923

3. Insert data into a SQL Lite database – create a table with the following data below that you will create yourself (Hint on how to create the SQL: *Python for Data Analysis 2nd edition* page 191, *Python for Data Analysis 3rd Edition:*Page 199):

a. Name, Address, City, State, Zip, Phone Number

b. Add at least 10 rows of data and submit your code with a query generating your results.