Question 2: Use Webscraping to Extract Tesla Revenue Data

Use the `requests` library to download the webpage https://www.macrotrends.net/stocks/charts/TSLA/tesla/revenue. Save the text of the response as a variable named `html_data`.

```
url= "https://www.macrotrends.net/stocks/charts/TSLA/tesla/revenue"
html_data=requests.get(url).text
```

Parse the html data using `beautiful_soup`.

```
soup = BeautifulSoup(html_data,"html5lib")
```

Using beautiful soup extract the table with `Tesla Quarterly Revenue` and store it into a dataframe named `tesla_revenue`. The dataframe should have columns `Date` and `Revenue`. Make sure the comma and dollar sign is removed from the `Revenue` column.

```
tesla_revenue= pd.read_html(url, match="Tesla Quarterly Revenue", flavor='bs4')[0]
tesla_revenue=tesla_revenue.rename(columns = {'Tesla Quarterly Revenue(Millions of US $)': 'Date', 'Tesla Quarterly Revenue(
tesla_revenue["Revenue"] = tesla_revenue["Revenue"].str.replace(",","").str.replace("$","")
tesla_revenue.head()
```

	Date	Revenue
0	2020-12-31	10744
1	2020-09-30	8771
2	2020-06-30	6036
3	2020-03-31	5985
4	2019-12-31	7384

Click here if you need help removing the dollar sign and comma

```
If you parsed the HTML table by row and column you can use the replace function on the string

    revenue = col[1].text.replace("$", "").replace(",", "")

If you use the read_html function you can use the replace function on the string representation of the
column

    tesla_revenue["Revenue"] = tesla_revenue["Revenue"].str.replace("$", "").str.replace(",", "")
```

Remove the rows in the dataframe that are empty strings or are NaN in the Revenue column. Print the entire `tesla_revenue` DataFrame to see if you have any.

```
tesla_revenue
```

	Date	Revenue
0	2020-12-31	10744
1	2020-09-30	8771
2	2020-06-30	6036
3	2020-03-31	5985
4	2019-12-31	7384
5	2019-09-30	6303
6	2019-06-30	6350
7	2019-03-31	4541
8	2018-12-31	7226
9	2018-09-30	6824
10	2018-06-30	4002
11	2018-03-31	3409
12	2017-12-31	3288
13	2017-09-30	2985
14	2017-06-30	2790

15	2017-03-31	2696
16	2016-12-31	2285
17	2016-09-30	2298
18	2016-06-30	1270
19	2016-03-31	1147
20	2015-12-31	1214
21	2015-09-30	937
22	2015-06-30	955
23	2015-03-31	940
24	2014-12-31	957
25	2014-09-30	852
26	2014-06-30	769
27	2014-03-31	621
28	2013-12-31	615
29	2013-09-30	431
30	2013-06-30	405
31	2013-03-31	562
32	2012-12-31	306
33	2012-09-30	50
34	2012-06-30	27
35	2012-03-31	30
36	2011-12-31	39
37	2011-09-30	58
38	2011-06-30	58
39	2011-03-31	49
40	2010-12-31	36
41	2010-09-30	31
42	2010-06-30	28
43	2010-03-31	21
44	2009-12-31	NaN
45	2009-09-30	46
46	2009-06-30	27
47	2008-12-31	NaN

Click here if you need help removing the Nan or empty strings

```
If you have NaN in the Revenue column

    tesla_revenue.dropna(inplace=True)

If you have emtpty string in the Revenue column
```

```
    tesla_revenue = tesla_revenue[tesla_revenue['Revenue'] != ""]
```

Display the last 5 row of the `tesla_revenue` dataframe using the `tail` function. Take a screenshot of the results.

```
tesla_revenue.dropna(inplace=True)
tesla_revenue.tail()
```

	Date	Revenue
41	2010-09-30	31
42	2010-06-30	28
43	2010-03-31	21
45	2009-09-30	46
46	2009-06-30	27