

```
GameStop_revenue_list = []

for row in soup.find("tbody").find_all("tr"):
    col = row.find_all("td")
    date = col[0].text
    revenue = col[1].text

    GameStop_revenue_list.append({"Date":date, "Revenue":revenue})
Gme_revenue = pd.DataFrame(GameStop_revenue_list)
Gme_revenue["Revenue"] = Gme_revenue["Revenue"].str.replace('$','').str.replace(',','')
```

[15] ✓ 0.0s Python

Display the last five rows of the `gme_revenue` dataframe using the `tail` function. Take a screenshot of the results.

```
Gme_revenue.tail(5)
```

[16] ✓ 0.0s Python

...

	Date	Revenue
11	2009	8806
12	2008	7094
13	2007	5319
14	2006	3092
15	2005	1843

```
Gme_revenue.dropna(inplace=True)

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

[17] ✓ 0.0s Python

Question 4: Use Webscraping to Extract GME Revenue Data

Use the `requests` library to download the webpage <https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html>. Save the text of the response as a variable named `html_data`.

```
[13] url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html'
      html_data=requests.get(url).text
      ✓ 4.7s Python
```

Parse the html data using `beautiful_soup`.

```
[14] soup=BeautifulSoup(html_data)
      ✓ 0.1s Python
```

Using `BeautifulSoup` or the `read_html` function extract the table with `GameStop Revenue` and store it into a dataframe named `gme_revenue`. The dataframe should have columns `Date` and `Revenue`. Make sure the comma and dollar sign is removed from the `Revenue` column using a method similar to what you did in Question 2.

▶ Click here if you need help locating the table

```
GameStop_revenue_list = []

for row in soup.find("tbody").find_all("tr"):
    col = row.find_all("td")
    date = col[0].text
    revenue = col[1].text
```