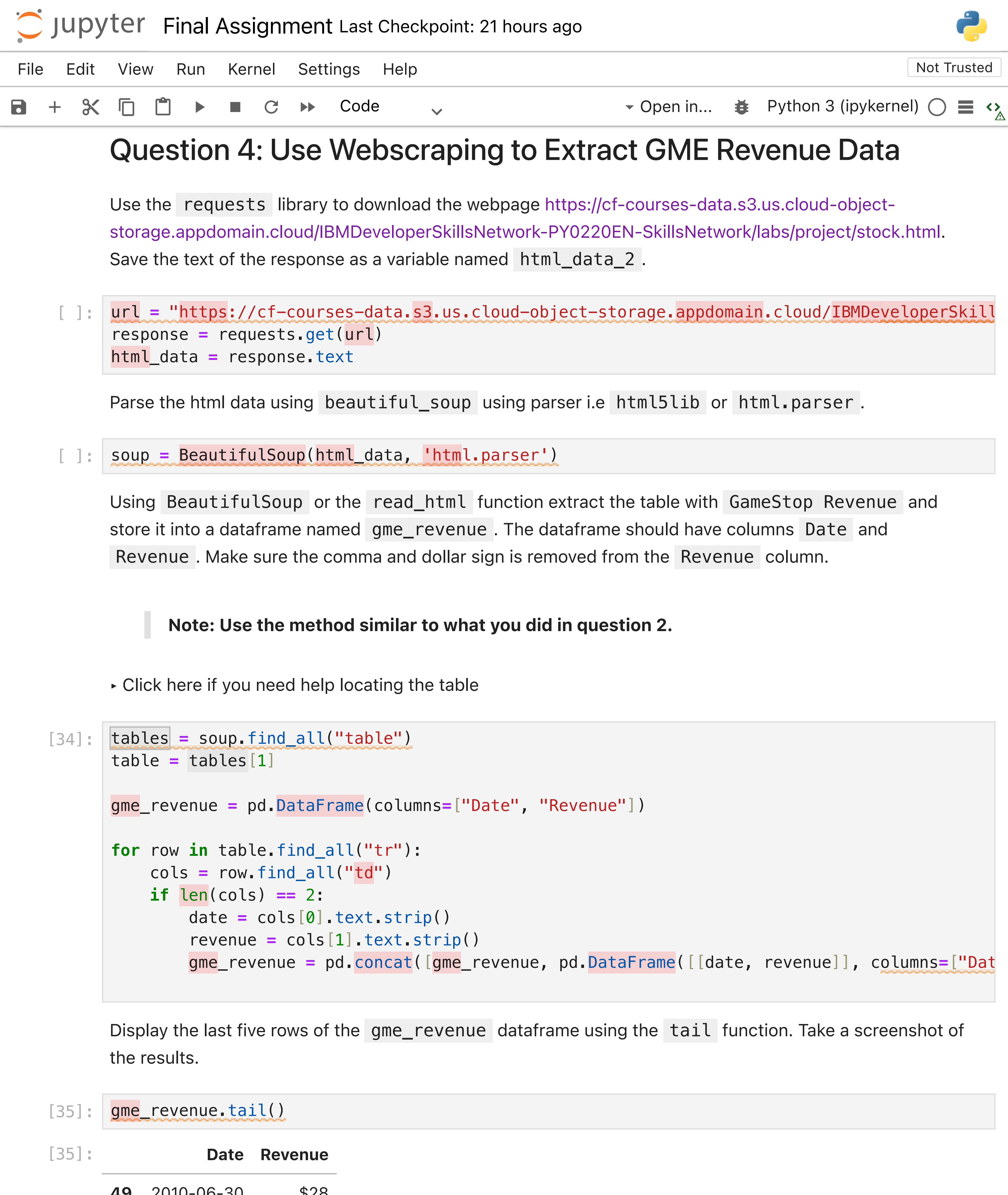

Question 4: Use Webscraping to Extract GME Revenue Data

Use the `requests` library to download the webpage https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-PY0220EN-SkillsNetwork/labs/project/stock.html. Save the text of the response as a variable named `html_data_2`.

[ ]:

```
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkill
response = requests.get(url)
html_data = response.text
```

Parse the html data using `beautiful_soup` using parser i.e `html5lib` or `html.parser`.

[ ]:

```
soup = BeautifulSoup(html_data, 'html.parser')
```

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.

> **Note: Use the method similar to what you did in question 2.**

▸ Click here if you need help locating the table

[34]:

```
tables = soup.find_all("table")
table = tables[1]

gme_revenue = pd.DataFrame(columns=["Date", "Revenue"])

for row in table.find_all("tr"):
    cols = row.find_all("td")
    if len(cols) == 2:
        date = cols[0].text.strip()
        revenue = cols[1].text.strip()
        gme_revenue = pd.concat([gme_revenue, pd.DataFrame([[date, revenue]], columns=["Dat
```

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

[35]:

```
gme_revenue.tail()
```

[35]:

```
          Date  Revenue
49  2010-06-30      $28
```