In [142…

```
'''
This Script uses the requests library along with the Beautiful Soup
Library to scrape the winning Numbers from the MegaMillions lottery
website. The resulting HTML is parsed and cleaned to pull the
winning numbers, megaplier, and drawing date from every drawing since
2003. Final results are then displayed as a pandas data frame.
The pandas dataframe can be inserted into an SQL database, exported to CSV,
or used with a library like matplotlib to perform analysis within the python
notebook.
'''
import requests
from bs4 import BeautifulSoup
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

#Create URL
URL = 'https://www.texaslottery.com/export/sites/lottery/Games/Mega_Millions/Winnin
page = requests.get(URL)

#create initial html object for parsing
soup = BeautifulSoup(page.content, 'html.parser')
find_class = soup.find_all('td')

#extract only the text within <td></td> and append to list object
results = []
for i in find_class:
    result = i.get_text()
    results.append(result)

#remove unwanted strings and empty strings from list
for i in results:
    if 'Million' in i or 'Billion' in i:
        results.remove(i)
for i in results:
    if 'CVO' in i:
        results.remove(i)
for i in results:
    if 'AP' in i:
        results.remove(i)
for i in results:
    if i.isspace():
       results.remove(i)

#remove every 5th string from the list, it is not needed
results_filtered = [string for i, string in enumerate(results, start=1) if i % 5 !=

#change format of winning numbers from 'x-x-x-x-x' to ['x','x','x','x','x']
results_cleaned = []
for i in results_filtered:
    results_cleaned.append(i.split('-'))

results_list_joined = []
for i in results_cleaned:
    results_list_joined.extend(i)

#seperate the drawing dates from the winning numbers
dates = []
winning_numbers = []
for i in results_list_joined:
    if '/' in i:
        dates.append(i)
    else:
        winning_numbers.append(int(i.strip()))

#iterate over all winning numbers,
#then put them into list object, 7 numbers at a time
chunk_size = 7
chunks = []
for i in range(0, len(winning_numbers), chunk_size):
    chunk = winning_numbers[i:i + chunk_size]
    chunks.append(chunk)

#turn drawing dates and corresponding winning numbers into a dictionary object
winning_numbers_dict = {}
index = 0
for i in dates:
    temp_dict = { i : chunks[index] }
    index += 1
    winning_numbers_dict.update(temp_dict)


#create new dictionary object formatted with keys as column names
#so that it can be turned into pandas dataframe
winning_numbers_data = {'Ball1': [],
        'Ball2': [],
        'Ball3': [],
        'Ball4': [],
        'Ball5': [],
        'MegaBall': [],
        'MegaPlier': [],
        'DrawingDate': []}

#take dates and winning numbers from first dictionary(winning_numbers_dict)
#and put them into the newly formatted dictionary(winning_numbers_data)
for key, value in winning_numbers_dict.items():
    index = 0
    for list in winning_numbers_data.values():
        if index < 7:
            list.append(value[index])
        else:
            list.append(key)
        index += 1

#create pandas dataframe
df = pd.DataFrame(winning_numbers_data)

df
```

Out[142…

	Ball1	Ball2	Ball3	Ball4	Ball5	MegaBall	MegaPlier	DrawingDate
0	52	60	61	66	67	23	4	12/03/2024
1	3	29	34	37	38	17	2	11/29/2024
2	5	22	24	39	42	3	3	11/26/2024
3	13	20	26	32	65	2	2	11/22/2024
4	5	35	50	51	59	8	4	11/19/2024
...	...	...	...	...	...	...	...	...
2186	2	13	21	22	49	52	4	12/23/2003
2187	5	10	17	35	39	38	3	12/19/2003
2188	16	24	31	46	47	47	3	12/16/2003
2189	4	14	15	24	48	41	4	12/09/2003
2190	1	12	15	18	44	42	4	12/05/2003

2191 rows × 8 columns