Data Visualization - PS4

Susan Zhang

2026-02-07

Due 02/07 at 5:00PM Central.

"This submission is my work alone and complies with the 30538 integrity policy." Add your initials to indicate your agreement: SZ

Github Classroom Assignment Setup and Submission Instructions

1. **Accepting and Setting up the PS4 Assignment Repository**
    - Each student must individually accept the repository for the problem set from Github Classroom ("ps4") – https://classroom.github.com/a/hWhtcHqH
        - You will be prompted to select your cnetid from the list in order to link your Github account to your cnetid.
        - If you can't find your cnetid in the link above, click "continue to next step" and accept the assignment, then add your name, cnetid, and Github account to this Google Sheet and we will manually link it: https://rb.gy/9u7fb6
    - If you authenticated and linked your Github account to your device, you should be able to clone your PS4 assignment repository locally.
    - Contents of PS4 assignment repository:
        - `ps4_template.qmd`: this is the Quarto file with the template for the problem set. You will write your answers to the problem set here.
2. **Submission Process**:
    - Knit your completed solution `ps4.qmd` as a pdf `ps4.pdf`.
        - Your submission does not need runnable code. Instead, you will tell us either what code you ran or what output you got.
    - To submit, push `ps4.qmd` and `ps4.pdf` to your PS4 assignment repository. Confirm on Github.com that your work was successfully pushed.

Grading

- You will be graded on what was last pushed to your PS4 assignment repository before the assignment deadline
- Problem sets will be graded for completion as: {missing (0%); - (incomplete, 50%); + (excellent, 100%)}
  - The percent values assigned to each problem denote how long we estimate the problem will take as a share of total time spent on the problem set, not the points they are associated with.
- In order for your submission to be considered complete, you need to push both your `ps4.qmd` and `ps4.pdf` to your repository. Submissions that do not include both files will automatically receive 50% credit.

```
import pandas as pd
import altair as alt
import time

import warnings
warnings.filterwarnings('ignore')
alt.renderers.enable("png")
```

```
RendererRegistry.enable('png')
```

Step 1: Develop initial scraper and crawler

```
import requests
from bs4 import BeautifulSoup

url = "https://oig.hhs.gov/fraud/enforcement/"
response = requests.get(url)
soup = BeautifulSoup(response.content, 'lxml')

news_ul = soup.find("ul", class_="usa-card-group padding-y-0")
news_li = news_ul.find_all("li", class_="usa-card card--list
 ↪  pep-card--minimal mobile:grid-col-12")

data = []
for li in news_li:
  title_a = li.find("h2", class_="usa-card__heading").find("a")
  title = title_a.get_text(strip=True)
  link = "https://oig.hhs.gov" + title_a["href"]

  date_span = li.find("div", class_="font-body-sm").find("span")
  date = date_span.get_text(strip=True)

  category_li = li.find("ul", class_="display-inline
 ↪  add-list-reset").find("li")
  category = category_li.get_text(strip=True)

  data.append({
    'Title': title,
    'Date': date,
    'Category': category,
    'Link': link
```

```
    })

df = pd.DataFrame(data)
df.head()
```

	Title	Date	Category	Lin
0	Houston Transplant Doctor Indicted For Making ...	February 5, 2026	Criminal and Civil Actions	htt
1	MultiCare Health System to Pay Millions to Set...	February 4, 2026	Criminal and Civil Actions	htt
2	Brooklyn Banker Pleads Guilty to Laundering Pr...	February 3, 2026	COVID-19	htt
3	Delafield Man Sentenced to 18 Months' Imprison...	February 3, 2026	Criminal and Civil Actions	htt
4	Former NFL Player Convicted for $197M Medicare...	February 3, 2026	Criminal and Civil Actions	htt

Step 2: Making the scraper dynamic

1. Turning the scraper into a function

- a. Pseudo-Code DEFINE FUNCTION scrape(start_month, start_year, run_scraper = False)

IF run_scraper IS False: RETURN

IF start_year < 2013: PRINT "Start year must be 2013 or later" RETURN

data <- empty list page <- 1 continue_crawling <- True

WHILE continue_crawling IS True:

```
url <- "https://oig.hhs.gov/fraud/enforcement/?page=" + page
page_content <- send request to url with headers
li_items <- parse page_content to find all news list items

IF li_items IS empty:
    continue_crawling <- False
    BREAK

FOR EACH item IN li_items:

    title <- extract title from item
    date_string <- extract date from item
    category <- extract category from item
    link <- extract link from item
```

```
    date <- convert date_string to datetime format

    IF date >= (start_year, start_month):
        APPEND (title, date, category, link) TO data
    ELSE:
        continue_crawling <- False
        BREAK OUT OF ALL LOOPS

WAIT 1 second
page <- page + 1
```

END WHILE

df <- convert data to DataFrame filename <- “enforcement_actions_” + start_year + “_” + start_month + “.csv” SAVE df AS filename

RETURN df, total number of records, earliest date

- b. Create Dynamic Scraper

```python
from datetime import datetime

def scrape(start_month, start_year, run_scraper=False):
  if not run_scraper:
    return pd.DataFrame()
  if start_year < 2013:
    print("Please restrict to year >= 2013, since only enforcement actions
     ↪  after 2013 are listed.")
    return pd.DataFrame()

  base_url = "https://oig.hhs.gov/fraud/enforcement/"
  headers = {"User-Agent": "Mozilla/5.0"}
  data = []
  page = 1
  continue_crawling = True
  start_date = datetime(start_year, start_month, 1)

  print(f"Start collecting the enforcement actions from {start_year} -
   ↪  {start_month}.")

  while continue_crawling:
    current_url = f"{base_url}?page={page}"
    try:
      response = requests.get(current_url, headers=headers)
```

```
    response.raise_for_status()
    soup = BeautifulSoup(response.content, "html.parser")

    news_ul = soup.find("ul", class_="usa-card-group padding-y-0")
    if not news_ul:
      continue_crawling = False
      break

    news_li = news_ul.find_all("li", class_="usa-card card--list
↪ pep-card--minimal mobile:grid-col-12")
    if not news_li:
      continue_crawling = False
      break

    for li in news_li:
      title_a = li.find("h2", class_="usa-card__heading").find("a")
      title = title_a.get_text(strip=True)
      link = "https://oig.hhs.gov" + title_a["href"]

      date_span = li.find("div", class_="font-body-sm").find("span")
      date_str = date_span.get_text(strip=True)
      try:
        item_date = datetime.strptime(date_str, "%B %d, %Y")
      except:
        item_date = None
        continue

      if item_date < start_date:
        continue_crawling = False
        break

      category_li = li.find("ul", class_="display-inline
↪ add-list-reset").find("li")
      category = category_li.get_text(strip=True)

      data.append({
        "Title": title,
        "Date": date_str,
        "Category": category,
        "Link": link,
        "Date_Datetime": item_date
        })
```

```
      time.sleep(1)
      print(f"Page {page} has been scraped, the total number of data is
       ↪  {len(data)} items.")
      page += 1

    except Exception as e:
      print(f"Error occurred while scraping page {page} {e}")
      continue_crawling = False
      break

  df = pd.DataFrame(data)
  if not df.empty:
    df = df.sort_values("Date_Datetime",
↪  ascending=False).reset_index(drop=True)
    csv_filename = f"enforcement_actions_{start_year}_{start_month}.csv"
    df.to_csv(csv_filename, index=False, encoding="utf-8")
    print(f"\nScraping completed, obtained {len(df)} pieces of data in total.
     ↪  The earliest data date is {df['Date'].iloc[-1]}. The data has been
     ↪  saved to {csv_filename}")
  else:
    print("\nNo data scraped.")

  return df

df_2024_01 = scrape(start_month=1, start_year=2024, run_scraper=True)
df_2024_01 = pd.read_csv("enforcement_actions_2024_1.csv")

print(f"Total enforcement actions (2024-01 to present): {len(df_2024_01)}")
df_2024_01["Date_Datetime"] = pd.to_datetime(df_2024_01["Date"], format="%B
 ↪  %d, %Y")
df_2024_01_sorted = df_2024_01.sort_values("Date_Datetime")
earliest_2024 = df_2024_01_sorted.iloc[0]
print(f"Earliest enforcement action date: {earliest_2024['Date']}")
```

```
Start collecting the enforcement actions from 2024 - 1.
Page 1 has been scraped, the total number of data is 20 items.
Page 2 has been scraped, the total number of data is 40 items.
Page 3 has been scraped, the total number of data is 60 items.
Page 4 has been scraped, the total number of data is 80 items.
Page 5 has been scraped, the total number of data is 100 items.
Page 6 has been scraped, the total number of data is 120 items.
Page 7 has been scraped, the total number of data is 140 items.
Page 8 has been scraped, the total number of data is 160 items.
```

```
Page 9 has been scraped, the total number of data is 180 items.
Page 10 has been scraped, the total number of data is 200 items.
Page 11 has been scraped, the total number of data is 220 items.
Page 12 has been scraped, the total number of data is 240 items.
Page 13 has been scraped, the total number of data is 260 items.
Page 14 has been scraped, the total number of data is 280 items.
Page 15 has been scraped, the total number of data is 300 items.
Page 16 has been scraped, the total number of data is 320 items.
Page 17 has been scraped, the total number of data is 340 items.
Page 18 has been scraped, the total number of data is 360 items.
Page 19 has been scraped, the total number of data is 380 items.
Page 20 has been scraped, the total number of data is 400 items.
Page 21 has been scraped, the total number of data is 420 items.
Page 22 has been scraped, the total number of data is 440 items.
Page 23 has been scraped, the total number of data is 460 items.
Page 24 has been scraped, the total number of data is 480 items.
Page 25 has been scraped, the total number of data is 500 items.
Page 26 has been scraped, the total number of data is 520 items.
Page 27 has been scraped, the total number of data is 540 items.
Page 28 has been scraped, the total number of data is 560 items.
Page 29 has been scraped, the total number of data is 580 items.
Page 30 has been scraped, the total number of data is 600 items.
Page 31 has been scraped, the total number of data is 620 items.
Page 32 has been scraped, the total number of data is 640 items.
Page 33 has been scraped, the total number of data is 660 items.
Page 34 has been scraped, the total number of data is 680 items.
Page 35 has been scraped, the total number of data is 700 items.
Page 36 has been scraped, the total number of data is 720 items.
Page 37 has been scraped, the total number of data is 740 items.
Page 38 has been scraped, the total number of data is 760 items.
Page 39 has been scraped, the total number of data is 780 items.
Page 40 has been scraped, the total number of data is 800 items.
Page 41 has been scraped, the total number of data is 820 items.
Page 42 has been scraped, the total number of data is 840 items.
Page 43 has been scraped, the total number of data is 860 items.
Page 44 has been scraped, the total number of data is 880 items.
Page 45 has been scraped, the total number of data is 900 items.
Page 46 has been scraped, the total number of data is 920 items.
Page 47 has been scraped, the total number of data is 940 items.
Page 48 has been scraped, the total number of data is 960 items.
Page 49 has been scraped, the total number of data is 980 items.
Page 50 has been scraped, the total number of data is 1000 items.
Page 51 has been scraped, the total number of data is 1020 items.
```

```
Page 52 has been scraped, the total number of data is 1040 items.
Page 53 has been scraped, the total number of data is 1060 items.
Page 54 has been scraped, the total number of data is 1080 items.
Page 55 has been scraped, the total number of data is 1100 items.
Page 56 has been scraped, the total number of data is 1120 items.
Page 57 has been scraped, the total number of data is 1140 items.
Page 58 has been scraped, the total number of data is 1160 items.
Page 59 has been scraped, the total number of data is 1180 items.
Page 60 has been scraped, the total number of data is 1200 items.
Page 61 has been scraped, the total number of data is 1220 items.
Page 62 has been scraped, the total number of data is 1240 items.
Page 63 has been scraped, the total number of data is 1260 items.
Page 64 has been scraped, the total number of data is 1280 items.
Page 65 has been scraped, the total number of data is 1300 items.
Page 66 has been scraped, the total number of data is 1320 items.
Page 67 has been scraped, the total number of data is 1340 items.
Page 68 has been scraped, the total number of data is 1360 items.
Page 69 has been scraped, the total number of data is 1380 items.
Page 70 has been scraped, the total number of data is 1400 items.
Page 71 has been scraped, the total number of data is 1420 items.
Page 72 has been scraped, the total number of data is 1440 items.
Page 73 has been scraped, the total number of data is 1460 items.
Page 74 has been scraped, the total number of data is 1480 items.
Page 75 has been scraped, the total number of data is 1500 items.
Page 76 has been scraped, the total number of data is 1520 items.
Page 77 has been scraped, the total number of data is 1540 items.
Page 78 has been scraped, the total number of data is 1560 items.
Page 79 has been scraped, the total number of data is 1580 items.
Page 80 has been scraped, the total number of data is 1600 items.
Page 81 has been scraped, the total number of data is 1620 items.
Page 82 has been scraped, the total number of data is 1640 items.
Page 83 has been scraped, the total number of data is 1660 items.
Page 84 has been scraped, the total number of data is 1680 items.
Page 85 has been scraped, the total number of data is 1700 items.
Page 86 has been scraped, the total number of data is 1720 items.
Page 87 has been scraped, the total number of data is 1740 items.
Page 88 has been scraped, the total number of data is 1760 items.
Page 89 has been scraped, the total number of data is 1780 items.
Page 90 has been scraped, the total number of data is 1787 items.

Scraping completed, obtained 1787 pieces of data in total. The earliest data
date is January 3, 2024. The data has been saved to
enforcement_actions_2024_1.csv
```

```
Total enforcement actions (2024-01 to present): 1787
Earliest enforcement action date: January 3, 2024
```

- c. Test Your Code

```
df_2022_01 = scrape(start_month=1,start_year=2022, run_scraper=True)
df_2022_01 = pd.read_csv("enforcement_actions_2022_1.csv")

print(f"Total enforcement actions (2022-01 to present): {len(df_2022_01)}")
df_2022_01["Date_Datetime"] = pd.to_datetime(df_2022_01["Date"], format="%B
 ↪  %d, %Y")
df_2022_01_sorted = df_2022_01.sort_values("Date_Datetime")
earliest_2022 = df_2022_01_sorted.iloc[0]
print(f"Earliest enforcement action date: {earliest_2022['Date']}")
```

```
Start collecting the enforcement actions from 2022 - 1.
Page 1 has been scraped, the total number of data is 20 items.
Page 2 has been scraped, the total number of data is 40 items.
Page 3 has been scraped, the total number of data is 60 items.
Page 4 has been scraped, the total number of data is 80 items.
Page 5 has been scraped, the total number of data is 100 items.
Page 6 has been scraped, the total number of data is 120 items.
Page 7 has been scraped, the total number of data is 140 items.
Page 8 has been scraped, the total number of data is 160 items.
Page 9 has been scraped, the total number of data is 180 items.
Page 10 has been scraped, the total number of data is 200 items.
Page 11 has been scraped, the total number of data is 220 items.
Page 12 has been scraped, the total number of data is 240 items.
Page 13 has been scraped, the total number of data is 260 items.
Page 14 has been scraped, the total number of data is 280 items.
Page 15 has been scraped, the total number of data is 300 items.
Page 16 has been scraped, the total number of data is 320 items.
Page 17 has been scraped, the total number of data is 340 items.
Page 18 has been scraped, the total number of data is 360 items.
Page 19 has been scraped, the total number of data is 380 items.
Page 20 has been scraped, the total number of data is 400 items.
Page 21 has been scraped, the total number of data is 420 items.
Page 22 has been scraped, the total number of data is 440 items.
Page 23 has been scraped, the total number of data is 460 items.
Page 24 has been scraped, the total number of data is 480 items.
Page 25 has been scraped, the total number of data is 500 items.
Page 26 has been scraped, the total number of data is 520 items.
Page 27 has been scraped, the total number of data is 540 items.
```

```
Page 28 has been scraped, the total number of data is 560 items.
Page 29 has been scraped, the total number of data is 580 items.
Page 30 has been scraped, the total number of data is 600 items.
Page 31 has been scraped, the total number of data is 620 items.
Page 32 has been scraped, the total number of data is 640 items.
Page 33 has been scraped, the total number of data is 660 items.
Page 34 has been scraped, the total number of data is 680 items.
Page 35 has been scraped, the total number of data is 700 items.
Page 36 has been scraped, the total number of data is 720 items.
Page 37 has been scraped, the total number of data is 740 items.
Page 38 has been scraped, the total number of data is 760 items.
Page 39 has been scraped, the total number of data is 780 items.
Page 40 has been scraped, the total number of data is 800 items.
Page 41 has been scraped, the total number of data is 820 items.
Page 42 has been scraped, the total number of data is 840 items.
Page 43 has been scraped, the total number of data is 860 items.
Page 44 has been scraped, the total number of data is 880 items.
Page 45 has been scraped, the total number of data is 900 items.
Page 46 has been scraped, the total number of data is 920 items.
Page 47 has been scraped, the total number of data is 940 items.
Page 48 has been scraped, the total number of data is 960 items.
Page 49 has been scraped, the total number of data is 980 items.
Page 50 has been scraped, the total number of data is 1000 items.
Page 51 has been scraped, the total number of data is 1020 items.
Page 52 has been scraped, the total number of data is 1040 items.
Page 53 has been scraped, the total number of data is 1060 items.
Page 54 has been scraped, the total number of data is 1080 items.
Page 55 has been scraped, the total number of data is 1100 items.
Page 56 has been scraped, the total number of data is 1120 items.
Page 57 has been scraped, the total number of data is 1140 items.
Page 58 has been scraped, the total number of data is 1160 items.
Page 59 has been scraped, the total number of data is 1180 items.
Page 60 has been scraped, the total number of data is 1200 items.
Page 61 has been scraped, the total number of data is 1220 items.
Page 62 has been scraped, the total number of data is 1240 items.
Page 63 has been scraped, the total number of data is 1260 items.
Page 64 has been scraped, the total number of data is 1280 items.
Page 65 has been scraped, the total number of data is 1300 items.
Page 66 has been scraped, the total number of data is 1320 items.
Page 67 has been scraped, the total number of data is 1340 items.
Page 68 has been scraped, the total number of data is 1360 items.
Page 69 has been scraped, the total number of data is 1380 items.
Page 70 has been scraped, the total number of data is 1400 items.
```

```
Page 71 has been scraped, the total number of data is 1420 items.
Page 72 has been scraped, the total number of data is 1440 items.
Page 73 has been scraped, the total number of data is 1460 items.
Page 74 has been scraped, the total number of data is 1480 items.
Page 75 has been scraped, the total number of data is 1500 items.
Page 76 has been scraped, the total number of data is 1520 items.
Page 77 has been scraped, the total number of data is 1540 items.
Page 78 has been scraped, the total number of data is 1560 items.
Page 79 has been scraped, the total number of data is 1580 items.
Page 80 has been scraped, the total number of data is 1600 items.
Page 81 has been scraped, the total number of data is 1620 items.
Page 82 has been scraped, the total number of data is 1640 items.
Page 83 has been scraped, the total number of data is 1660 items.
Page 84 has been scraped, the total number of data is 1680 items.
Page 85 has been scraped, the total number of data is 1700 items.
Page 86 has been scraped, the total number of data is 1720 items.
Page 87 has been scraped, the total number of data is 1740 items.
Page 88 has been scraped, the total number of data is 1760 items.
Page 89 has been scraped, the total number of data is 1780 items.
Page 90 has been scraped, the total number of data is 1800 items.
Page 91 has been scraped, the total number of data is 1820 items.
Page 92 has been scraped, the total number of data is 1840 items.
Page 93 has been scraped, the total number of data is 1860 items.
Page 94 has been scraped, the total number of data is 1880 items.
Page 95 has been scraped, the total number of data is 1900 items.
Page 96 has been scraped, the total number of data is 1920 items.
Page 97 has been scraped, the total number of data is 1940 items.
Page 98 has been scraped, the total number of data is 1960 items.
Page 99 has been scraped, the total number of data is 1980 items.
Page 100 has been scraped, the total number of data is 2000 items.
Page 101 has been scraped, the total number of data is 2020 items.
Page 102 has been scraped, the total number of data is 2040 items.
Page 103 has been scraped, the total number of data is 2060 items.
Page 104 has been scraped, the total number of data is 2080 items.
Page 105 has been scraped, the total number of data is 2100 items.
Page 106 has been scraped, the total number of data is 2120 items.
Page 107 has been scraped, the total number of data is 2140 items.
Page 108 has been scraped, the total number of data is 2160 items.
Page 109 has been scraped, the total number of data is 2180 items.
Page 110 has been scraped, the total number of data is 2200 items.
Page 111 has been scraped, the total number of data is 2220 items.
Page 112 has been scraped, the total number of data is 2240 items.
Page 113 has been scraped, the total number of data is 2260 items.
```

```
Page 114 has been scraped, the total number of data is 2280 items.
Page 115 has been scraped, the total number of data is 2300 items.
Page 116 has been scraped, the total number of data is 2320 items.
Page 117 has been scraped, the total number of data is 2340 items.
Page 118 has been scraped, the total number of data is 2360 items.
Page 119 has been scraped, the total number of data is 2380 items.
Page 120 has been scraped, the total number of data is 2400 items.
Page 121 has been scraped, the total number of data is 2420 items.
Page 122 has been scraped, the total number of data is 2440 items.
Page 123 has been scraped, the total number of data is 2460 items.
Page 124 has been scraped, the total number of data is 2480 items.
Page 125 has been scraped, the total number of data is 2500 items.
Page 126 has been scraped, the total number of data is 2520 items.
Page 127 has been scraped, the total number of data is 2540 items.
Page 128 has been scraped, the total number of data is 2560 items.
Page 129 has been scraped, the total number of data is 2580 items.
Page 130 has been scraped, the total number of data is 2600 items.
Page 131 has been scraped, the total number of data is 2620 items.
Page 132 has been scraped, the total number of data is 2640 items.
Page 133 has been scraped, the total number of data is 2660 items.
Page 134 has been scraped, the total number of data is 2680 items.
Page 135 has been scraped, the total number of data is 2700 items.
Page 136 has been scraped, the total number of data is 2720 items.
Page 137 has been scraped, the total number of data is 2740 items.
Page 138 has been scraped, the total number of data is 2760 items.
Page 139 has been scraped, the total number of data is 2780 items.
Page 140 has been scraped, the total number of data is 2800 items.
Page 141 has been scraped, the total number of data is 2820 items.
Page 142 has been scraped, the total number of data is 2840 items.
Page 143 has been scraped, the total number of data is 2860 items.
Page 144 has been scraped, the total number of data is 2880 items.
Page 145 has been scraped, the total number of data is 2900 items.
Page 146 has been scraped, the total number of data is 2920 items.
Page 147 has been scraped, the total number of data is 2940 items.
Page 148 has been scraped, the total number of data is 2960 items.
Page 149 has been scraped, the total number of data is 2980 items.
Page 150 has been scraped, the total number of data is 3000 items.
Page 151 has been scraped, the total number of data is 3020 items.
Page 152 has been scraped, the total number of data is 3040 items.
Page 153 has been scraped, the total number of data is 3060 items.
Page 154 has been scraped, the total number of data is 3080 items.
Page 155 has been scraped, the total number of data is 3100 items.
Page 156 has been scraped, the total number of data is 3120 items.
```

```
Page 157 has been scraped, the total number of data is 3140 items.
Page 158 has been scraped, the total number of data is 3160 items.
Page 159 has been scraped, the total number of data is 3180 items.
Page 160 has been scraped, the total number of data is 3200 items.
Page 161 has been scraped, the total number of data is 3220 items.
Page 162 has been scraped, the total number of data is 3240 items.
Page 163 has been scraped, the total number of data is 3260 items.
Page 164 has been scraped, the total number of data is 3280 items.
Page 165 has been scraped, the total number of data is 3300 items.
Page 166 has been scraped, the total number of data is 3320 items.
Page 167 has been scraped, the total number of data is 3340 items.
Page 168 has been scraped, the total number of data is 3360 items.
Page 169 has been scraped, the total number of data is 3377 items.

Scraping completed, obtained 3377 pieces of data in total. The earliest data
date is January 4, 2022. The data has been saved to
enforcement_actions_2022_1.csv
Total enforcement actions (2022-01 to present): 3377
Earliest enforcement action date: January 4, 2022
```

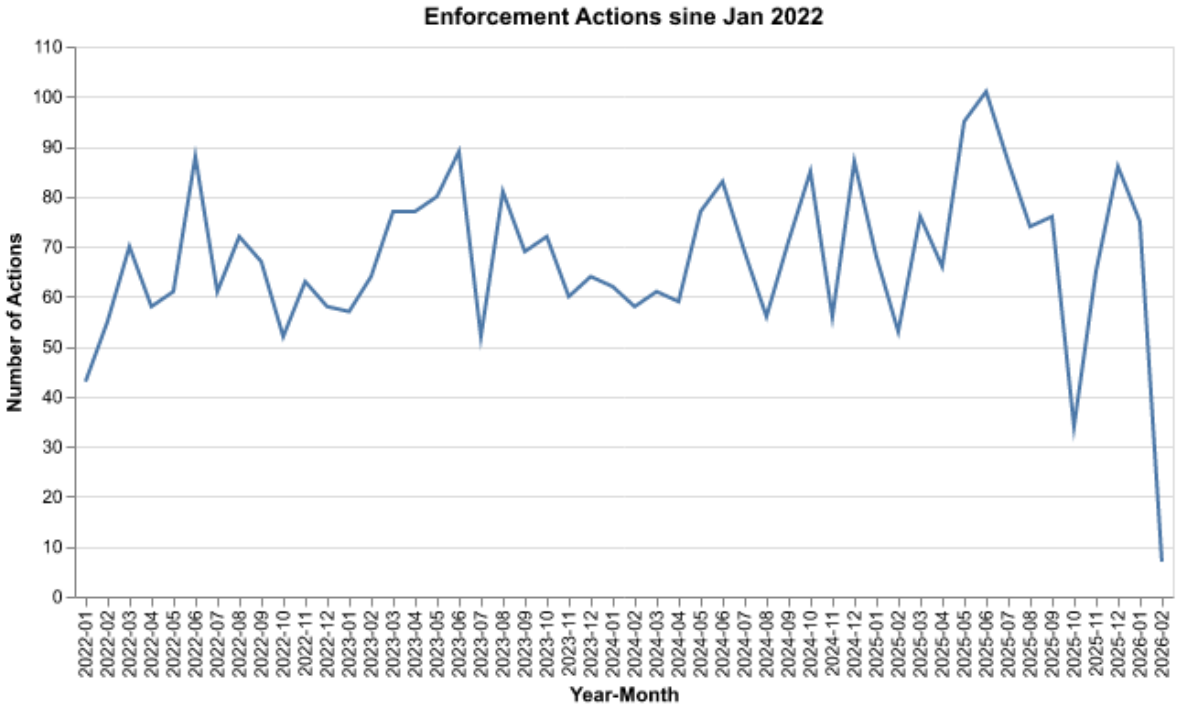
Step 3: Plot data based on scraped data

1. Plot the number of enforcement actions over time

```
df_2022_01["Date_Datetime"] = pd.to_datetime(df_2022_01["Date"], format="%B
 ↪  %d, %Y")
df_2022_01["Year_Month"] = df_2022_01["Date_Datetime"].dt.strftime("%Y-%m")

monthly_total = df_2022_01.groupby("Year_Month").agg(Count=("Title",
 ↪  "count")).reset_index()

chart1 = alt.Chart(monthly_total).mark_line().encode(
  alt.X("Year_Month:O", title="Year-Month"),
  alt.Y("Count:Q", title="Number of Actions")
).properties(title="Enforcement Actions sine Jan 2022", width=600,
 ↪  height=300)
chart1
```



2. Plot the number of enforcement actions categorized:

- based on "Criminal and Civil Actions" vs. "State Enforcement Agencies"

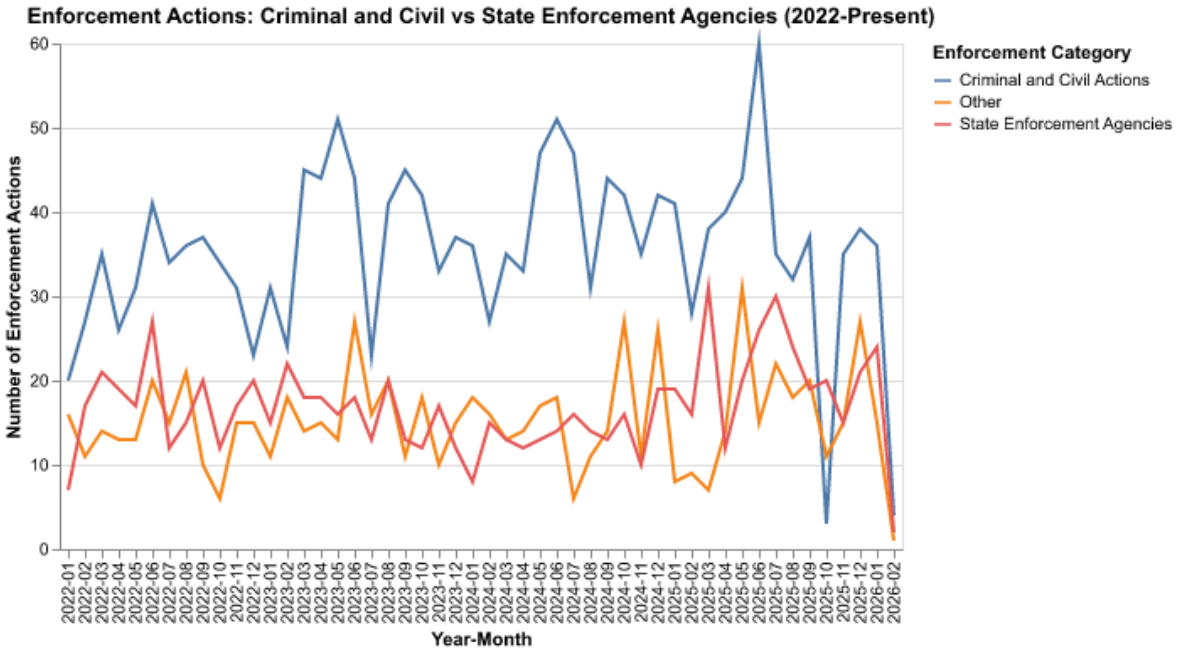
```
df_2022_01["Main_Category"] = df_2022_01["Category"].apply(
    lambda x: x if x in ["Criminal and Civil Actions", "State Enforcement
    ↪ Agencies"] else "Other"
)

def get_sub_category(title):
  title = title.lower()
  if "health care" in title or "medical" in title:
    return "Health Care Fraud"
  elif "financial" in title or "bank" in title:
    return "Financial Fraud"
  elif "drug" in title or "opioid" in title:
    return "Drug Enforcement"
  elif "bribery" in title or "corruption" in title:
    return "Bribery/Corruption"
  else:
    return "Other"
```

```
df_2022_01["Sub_Category"] = df_2022_01.apply(
    lambda row: get_sub_category(row["Title"]) if row["Main_Category"] ==
     ↪  "Criminal and Civil Actions" else None,
    axis=1
)

monthly_main = df_2022_01.groupby(["Year_Month", "Main_Category"]).agg(
    Action_Count=("Title", "count")
).reset_index()

chart2 = alt.Chart(monthly_main).mark_line().encode(
  alt.X("Year_Month:O", title="Year-Month"),
  alt.Y("Action_Count:Q", title="Number of Enforcement Actions"),
  alt.Color("Main_Category:N", title="Enforcement Category")
).properties(title="Enforcement Actions: Criminal and Civil vs State
 ↪  Enforcement Agencies (2022-Present)", width=500, height=300)
chart2
```



- based on five topics

```
df_criminal_civil = df_2022_01[df_2022_01["Main_Category"] == "Criminal and
 ↪  Civil Actions"].dropna(subset=["Sub_Category"])

monthly_sub = df_criminal_civil.groupby(["Year_Month", "Sub_Category"]).agg(
    Action_Count=("Title", "count")
).reset_index()

chart3 = alt.Chart(monthly_sub).mark_line().encode(
  alt.X("Year_Month:O", title="Year-Month"),
  alt.Y("Action_Count:Q", title="Number of Enforcement Actions"),
  alt.Color("Sub_Category:N", title="Sub Category")
).properties(title="Criminal and Civil Actions: Breakdown by Sub Category
 ↪  (2022-Present)", width=500, height=300)
chart3
```