PS5

Yuliana and Dale

Invalid Date

Due 11/9 at 5:00PM Central. Worth 100 points + 10 points extra credit.

Submission Steps (10 pts)

1. This problem set is a paired problem set.
2. Play paper, scissors, rock to determine who goes first. Call that person *Partner 1*.
   - Partner 1 (name and cnet ID): Yuliana Zhang ; yuejiu
   - Partner 2 (name and cnet ID): Dale (Yuanhao) Jin; jin86
3. Partner 1 will accept the `ps5` and then share the link it creates with their partner. You can only share it with one partner so you will not be able to change it after your partner has accepted.
4. "This submission is our work alone and complies with the 30538 integrity policy." Add your initials to indicate your agreement: ** YZ** ** Dale Jin**
5. "I have uploaded the names of anyone else other than my partner and I worked with on the problem set **No**" (1 point)
6. Late coins used this pset: ** 1 ** Late coins left after submission: ** 1 **
7. Knit your `ps5.qmd` to an PDF file to make `ps5.pdf`,
   - The PDF should not be more than 25 pages. Use `head()` and re-size figures when appropriate.
8. (Partner 1): push `ps5.qmd` and `ps5.pdf` to your github repo.
9. (Partner 1): submit `ps5.pdf` via Gradescope. Add your partner on Gradescope.
10. (Partner 1): tag your submission in Gradescope

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

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

import requests
from bs4 import BeautifulSoup
```

Step 1: Develop initial scraper and crawler

1. Scraping (PARTNER 1)

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

# Initialize lists to store extracted information
titles = []
dates = []
categories = []
links = []
# Find each enforcement action entry
for item in soup.find_all('li', class_='usa-card card--list pep-card--minimal
 ↪  mobile:grid-col-12'):
    # Extract title and link
    title_tag = item.find('h2', class_='usa-card__heading').find('a')
    title = title_tag.text.strip()
    link = title_tag['href']

    titles.append(title)
    links.append(f'https://oig.hhs.gov{link}')  # Form the full URL

    # Extract date
    date = item.find('span', class_='text-base-dark
 ↪  padding-right-105').text.strip()
    dates.append(date)
    # Extract category
    category_tag = item.find('ul', class_='display-inline add-list-reset').find('li')
    category = 'N/A'
    category = category_tag.text.strip()
    categories.append(category)
    data = pd.DataFrame({
    'Title': titles,
    'Date': dates,
    'Category': categories,
    'Link': links
})

# Display the head of the DataFrame
print(data.head())
```

```
                                               Title              Date  \
0  Macomb County Doctor And Pharmacist Agree To P...  November 4, 2024
1  Rocky Hill Pharmacy And Its Owners Indicted Fo...  November 4, 2024
2  North Texas Medical Center Pays $14.2 Million ...  November 4, 2024
3  New England Doctor Pleads Guilty To Drug Distr...  November 4, 2024
4  St. Louis County Woman Accused Of $3 Million H...  November 1, 2024

                     Category  \
0  Criminal and Civil Actions
1  Criminal and Civil Actions
2  Criminal and Civil Actions
3  Criminal and Civil Actions
4  Criminal and Civil Actions

                                                Link
0  https://oig.hhs.gov/fraud/enforcement/macomb-c...
1  https://oig.hhs.gov/fraud/enforcement/rocky-hi...
2  https://oig.hhs.gov/fraud/enforcement/north-te...
3  https://oig.hhs.gov/fraud/enforcement/new-engl...
4  https://oig.hhs.gov/fraud/enforcement/st-louis...
```

2. Crawling (PARTNER 1)

```
# Part 2: Adding the Agency Name by Crawling Each Link
agencies = []

# Loop through each enforcement action's detailed page
for link in data['Link']:
    response = requests.get(link)
    detail_soup = BeautifulSoup(response.content, 'lxml')
    agency_name = 'N/A'

    # Locate the <ul> tag containing the details
    details_list = detail_soup.find('ul', class_='usa-list usa-list--unstyled margin-y-2')
    if details_list:
        for li in details_list.find_all('li'):
            label_span = li.find('span', class_='padding-right-2 text-base')
            if label_span:
                label_text = label_span.text.strip()
                # Check if the label is "Date:" or "Agency:"
                if label_text == "Agency:":
                    agency_name = label_span.find_next_sibling(text=True).strip()
    # Append extracted data to lists
    agencies.append(agency_name)

# Add the date and agency names to the DataFrame
data['Agency'] = agencies

# Display the updated DataFrame
print(data.head())
```

```
                                                Title              Date  \
0  Macomb County Doctor And Pharmacist Agree To P...  November 4, 2024
1  Rocky Hill Pharmacy And Its Owners Indicted Fo...  November 4, 2024
2  North Texas Medical Center Pays $14.2 Million ...  November 4, 2024
3  New England Doctor Pleads Guilty To Drug Distr...  November 4, 2024
4  St. Louis County Woman Accused Of $3 Million H...  November 1, 2024

                     Category  \
0  Criminal and Civil Actions
1  Criminal and Civil Actions
2  Criminal and Civil Actions
```

```
3  Criminal and Civil Actions
4  Criminal and Civil Actions

                                               Link  \
0  https://oig.hhs.gov/fraud/enforcement/macomb-c...
1  https://oig.hhs.gov/fraud/enforcement/rocky-hi...
2  https://oig.hhs.gov/fraud/enforcement/north-te...
3  https://oig.hhs.gov/fraud/enforcement/new-engl...
4  https://oig.hhs.gov/fraud/enforcement/st-louis...

                                             Agency
0  U.S. Attorney's Office, Eastern District of Mi...
1  U.S. Attorney's Office, Eastern District of Te...
2  U.S. Attorney's Office, Northern District of T...
3                         U.S. Department of Justice
4  U.S. Attorney's Office, Eastern District of Mi...
```

Step 2: Making the scraper dynamic

1. Turning the scraper into a function

- a. Pseudo-Code (PARTNER 2)
- b. Create Dynamic Scraper (PARTNER 2)
- c. Test Partner's Code (PARTNER 1)

Step 3: Plot data based on scraped data

1. Plot the number of enforcement actions over time (PARTNER 2)

2. Plot the number of enforcement actions categorized: (PARTNER 1)

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

Step 4: Create maps of enforcement activity

1. Map by State (PARTNER 1)

2. Map by District (PARTNER 2)

Extra Credit

1. Merge zip code shapefile with population

2. Conduct spatial join

3. Map the action ratio in each district