Practice problems for the CS50P Week 4

p4libraries

Goals

- Write a Python program that retrieves public data in JSON format via a Web API, filters it, performs statistical calculations, and displays the results in a graph.
- Learn to use different Python libraries.

Steps

1. Visit **data.gov.lv** or **data.europa.eu** and choose a dataset that is of interest to you and available in JSON format.
2. Use the Python requests library to send a GET request to the chosen API endpoint. Make sure to check the API documentation for any required parameters. Example:

```
import requests

url = "API_ENDPOINT_HERE"
response = requests.get(url)
data = response.json()
```

3. Analyze the JSON structure. Utilize online resources or write Python code to format the output for better readability. You may need to use the Linux 'more' command if the output is lengthy (in terminal use ‘python file.py | more’, ‘Space’ for next page, ‘Q’ to quit). Example:

```
import json

print( json.dumps(data, indent=2) )
```

4. Convert the retrieved data into a pandas DataFrame. This step may require some preprocessing depending on the data's structure. Preferably, provide a list of dictionaries to the pd.DataFrame() function. Example:

```
import pandas as pd

df = pd.DataFrame(data) # Assuming `data` is a list of dictionaries
```

5. Perform your desired analysis. This could include cleaning the data, filtering, grouping, or any other statistical calculations. Example:

```
filtered_df = df.loc[df['Age'] > 30] # Filter by age
print(filtered_df.describe())    # Get a statistical summary
```

6. Use matplotlib to create a graph based on your analysis. This could be a line graph, bar chart, histogram, etc., depending on what best represents your findings. Example:

```
import matplotlib.pyplot as plt

plot(kind='bar')
plt.title('Ages')
plt.xlabel('Year')
plt.ylabel('Ages')
#plt.show()
plt.savefig('outout.png')
```

Submitting your code

```
submit50 okskola/cs50addp/main/p4libraries
```