In [1]:
```
pip install requests beautifulsoup4
```

```
Requirement already satisfied: requests in c:\users\engin\anaconda3\lib\s
ite-packages (2.31.0)
Requirement already satisfied: beautifulsoup4 in c:\users\engin\anaconda3
\lib\site-packages (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\engin
\anaconda3\lib\site-packages (from requests) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\engin\anaconda3\l
ib\site-packages (from requests) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\engin\anaco
nda3\lib\site-packages (from requests) (1.26.16)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\engin\anaco
nda3\lib\site-packages (from requests) (2023.7.22)
Requirement already satisfied: soupsieve>1.2 in c:\users\engin\anaconda3
\lib\site-packages (from beautifulsoup4) (2.4)
Note: you may need to restart the kernel to use updated packages.
```

In [ ]:

```
import requests
from bs4 import BeautifulSoup
import csv

url = "https://www.youtube.com/@PW-Foundation/videos"

# Send an HTTP request to the URL
response = requests.get(url)

# Check if the request was successful (status code 200)
if response.status_code == 200:
    # Parse the HTML content of the page
    soup = BeautifulSoup(response.text, 'html.parser')

    # Lists to store extracted data
    video_urls = []
    thumbnail_urls = []
    video_titles = []
    view_counts = []
    post_times = []

    # Extract data for the first five videos
    for video in soup.select('div.style-scope ytd-grid-video-renderer')[:5]
        # Extract video URL
        video_url = "https://www.youtube.com" + video.select_one('a#thumbna
        video_urls.append(video_url)

        # Extract video thumbnail URL
        thumbnail_url = video.select_one('img#img')['src']
        thumbnail_urls.append(thumbnail_url)

        # Extract video title
        video_title = video.select_one('yt-formatted-string#video-title').t
        video_titles.append(video_title)

        # Extract view count
        view_count = video.select_one('span#metadata-line span.style-scope
        view_counts.append(view_count)

        # Extract time of posting
        post_time = video.select_one('span#metadata-line yt-formatted-strin
        post_times.append(post_time)

    # Save the data to a CSV file
    with open('youtube_data.csv', 'w', newline='', encoding='utf-8') as csv
        fieldnames = ['Video URL', 'Thumbnail URL', 'Title', 'View Count',
        writer = csv.DictWriter(csvfile, fieldnames=fieldnames)

        writer.writeheader()
        for i in range(5):
            writer.writerow({'Video URL': video_urls[i],
                             'Thumbnail URL': thumbnail_urls[i],
                             'Title': video_titles[i],
                             'View Count': view_counts[i],
                             'Post Time': post_times[i]})
```

In [ ]: