SlateMate AI/ML Technical Assignment

🔹 Assignment 1: Social Media Data Scraper (with Thumbnail & Metadata Collection)

🧠 Background:

In order to power SlateMate's AI moderation and personalization engine, we need to collect **real-world content** from social media platforms. This includes captions, hashtags, timestamps, thumbnails, and engagement metrics from posts on platforms like Instagram, Facebook, and YouTube.

This assignment focuses on building a **scraper** that can extract this structured data and media content from public profiles, hashtags, or search results.

🎯 Objective:

Build a **cross-platform social media data scraper** that collects metadata and thumbnails/images from at least one major platform (preferably Instagram or YouTube).

📌 Scope of Work:

You are expected to:

1. **Identify a scraping method** that works for at least one social media platform (Instagram, Facebook, YouTube, Twitter/X).
2. Extract structured metadata fields from public posts.
3. Download and save the thumbnail or media image.
4. Store everything in a CSV and a local folder.
5. Ensure that scraping works for at least 30–50 posts reliably.

📥 Input Requirements:

The script should allow:

- A target (hashtag, public profile URL, or search term)
- The platform to be selected (`instagram`, `youtube`, etc.)

Example:

bash
CopyEdit

```
python scrape_posts.py --platform instagram --target "#chess"
```

🗂️ Expected Output Files:

✅ 1. `metadata.csv`:

A structured CSV file with the following fields:

Field	Description
`post_id`	Unique post identifier
`platform`	e.g., Instagram, YouTube
`post_text`	Caption or description text
`hashtags`	List of hashtags in the post
`timestamp`	When the post was created
`image_url`	Direct URL or file name of thumbnail
`likes`	(Optional) Number of likes
`comments`	(Optional) Number of comments
`author`	Profile or page name (if available)

✅ 2. `thumbnails/`:

- A folder containing all downloaded images or thumbnails named as `<post_id>.jpg`

🔧 Tools & Techniques (Hints):

📷 Instagram:

- Use **Selenium** or **Playwright** to automate scrolling and DOM reading
- Extract metadata from the loaded HTML or inline JSON
- Use developer tools to locate image URLs

📺 YouTube:

- Use YouTube **Data API v3** (official)
- Get titles, video thumbnails, publish dates, channel info
- Requires an API key (free from Google Cloud)

🐦 Twitter/X:

- Use `snscrape` or a combination of BeautifulSoup + Tweepy
- Limited access, so prefer scraping public search results

⚠️ Requirements:

- Respect platform terms of service (scrape only public profiles)
- Include retry & timeout handling
- Comment code clearly
- Store everything in UTF-8 format

📦 Final Deliverables (Required for Submission):

1. `scrape_posts.py` or Jupyter notebook
2. `metadata.csv` with at least 50 rows
3. `thumbnails/` folder with all thumbnails
4. `README.md` explaining:
    - Which platform you scraped
    - Challenges faced and how you solved them
    - How to run the script

🏆 Bonus (Optional but Appreciated):

- Multi-platform support (e.g., scrape both YouTube & Instagram)

- Add browser headless mode and page scroll detection
- Add keyword-based filtering (e.g., skip posts with blocked keywords)
- Add CLI progress bar or logging

✅ Evaluation Criteria:

Component	Weight
Completeness of Metadata	25%
Scraping Accuracy & Coverage	25%
Code Structure & Modularity	20%
Thumbnail Handling & Organization	10%
Error Handling & CLI Usability	10%
README Quality	10%