Assignment: Telegram .onion Link Extractor

Objective:

Build a small Python script that connects to a Telegram channel using the **Telegram API** and extracts any `.onion` links from recent messages.

Requirements:

1. **Use the Telethon library** to connect to the Telegram API.

2. Monitor **the public Telegram channel** toronionlinks (or any other similar public channel) for **.onion** links.

3. Extract all messages using **Telethon**.

4. Parse the messages using **regex** to extract any `.onion` URLs.

5. Save the extracted information in the following **JSON format** (one JSON object per line):

```
{
  "source": "telegram",
  "url": "http://abcd1234xyz.onion",
  "discovered_at": "2025-05-12T10:00:00Z",
  "context": "Found in Telegram channel @toronionlinks",
  "status": "pending"
}
```

6. Output the results into a file called `onion_links.json`.

Bonus (Optional):

- Store the **last message ID** in a text file and skip older messages on the next run (simulate deduplication).
- Use **async/await** correctly if they implement asynchronous code.
- Handle rate limits or errors gracefully (e.g., API failures, network issues).

Setup Hints for Candidates:

1. Create a free Telegram account and apply for **API credentials** at my.telegram.org.
2. Install Telethon: `pip install telethon`
3. You can use **@**toronionlinks (or any other public channel) for extracting `.onion` links, or test with a dummy channel of your own.

This assignment is **simple but comprehensive**, checking skills in **web scraping, API integration, regex, JSON handling**, and **modular coding**.