import praw

import time

import random

# Set up Reddit API credentials

reddit = praw.Reddit(

client\_id="Lo2\_zCPNEBnYxhghn\_gQiQ", # Your Client ID

client\_secret="PLwoo0YegVibgDLLG86bSzYq-gZiXQ", # Your Client Secret

username="QuantamHack", # Your Reddit bot username

password="Olafunmi2018!", # Replace with your Reddit account password

user\_agent="QuantamHackBot by u/QuantamHack - A bot to explore sci-fi threads"

)

# Fetch the subreddits your account has joined

print("🔍 Fetching subreddits your account has joined...")

subscribed\_subreddits = [sub.display\_name for sub in reddit.user.subreddits(limit=None)]

print(f"✨ Subscribed to {len(subscribed\_subreddits)} subreddits: {', '.join(subscribed\_subreddits)}")

# Keywords and personalized responses

keywords\_and\_responses = {

"elon musk": "Speaking of Elon Musk, his work inspires many aspects of my sci-fi story, \*Bytes of the Matrix\*. It's all about AI, consciousness, and dystopian futures. Check it out here: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

"neuralink": "Neuralink's advancements remind me of the themes I explore in \*Bytes of the Matrix\*. It's a story about the future of technology and human consciousness. If you're into sci-fi, here's the link: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

"AI": "AI is such a fascinating topic! It’s central to my sci-fi story, \*Bytes of the Matrix\*. If you’re interested, check it out here: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

"cyberpunk": "Your post feels like it could be a scene straight out of \*Bytes of the Matrix\*! My story dives deep into cyberpunk themes. Take a look: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

"dystopia": "Dystopian futures inspire so much creativity, including my story \*Bytes of the Matrix\*. Check it out here: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

"machine learning": "Machine learning is so transformative. It's a key concept in my story, \*Bytes of the Matrix\*. Feel free to check it out: [https://substack.com/@codedvision](https://substack.com/@codedvision).",

# Add more keywords and responses as needed

}

# Default comment for posts that don't match specific keywords

default\_comment = (

"Hi! I noticed your post and wanted to share my sci-fi story, \*Bytes of the Matrix\*. It's all about AI, consciousness, and dystopian futures. If you're curious, check it out here: [https://substack.com/@codedvision](https://substack.com/@codedvision)."

)

# Create or open a log file

log\_file = open("bot\_activity\_log.txt", "a")

# Loop through the subscribed subreddits

for sub in subscribed\_subreddits:

subreddit = reddit.subreddit(sub)

print(f"🌌 Searching for relevant posts in r/{sub}...")

for submission in subreddit.new(limit=10): # Fetch the 10 newest posts

# Check if the post title or selftext contains any of the keywords

matched\_keyword = next(

(keyword for keyword in keywords\_and\_responses if keyword.lower() in submission.title.lower() or keyword.lower() in submission.selftext.lower()),

None

)

if matched\_keyword:

print(f"🎯 Found a relevant post: {submission.title} (Keyword: {matched\_keyword})")

# Check if the bot has already commented on this post

submission.comments.replace\_more(limit=0) # Ensure we load all top-level comments

already\_commented = any(comment.author == "QuantamHack" for comment in submission.comments)

if not already\_commented:

try:

# Select the personalized response for the matched keyword

comment\_text = keywords\_and\_responses.get(matched\_keyword, default\_comment)

# Post the comment

submission.reply(comment\_text)

print(f"✅ Commented on the post: {submission.title}")

# Log the activity

log\_file.write(f"Commented on: {submission.title} (URL: {submission.url})\n")

# Add a delay to avoid spamming

print("⏳ Waiting 60 seconds before continuing...")

time.sleep(60) # Wait 60 seconds

except Exception as e:

print(f"❌ Error posting comment: {e}")

log\_file.write(f"Error commenting on: {submission.title} (Error: {e})\n")

else:

print("🤖 Bot has already commented on this post.")

else:

print(f"🚫 No relevant keywords found in: {submission.title}")

log\_file.write(f"Skipped: {submission.title} (URL: {submission.url})\n")

# Close the log file

log\_file.close()

print("✅ Bot has finished running.")