from flask import Flask, request, render\_template, redirect, url\_for

import google.generativeai as genai

app = Flask(\_\_name\_\_)

API\_KEY = "AIzaSyDS7qQoeHhjU0ezBIN2cPWmx545Ngi9Xfg"

def initialize\_model(api\_key):

    try:

        genai.configure(api\_key=api\_key)

        generation\_config = {

            "temperature": 0.7,

            "top\_p": 0.95,

            "top\_k": 40,

            "max\_output\_tokens": 8192,

        }

        #*DIFFERENT MODELS*

        return genai.GenerativeModel(model\_name="gemini-2.0-flash-lite", generation\_config=generation\_config)

        #*return genai.GenerativeModel(model\_name="gemini-1.5-flash-8b", generation\_config=generation\_config)*

        #*return genai.GenerativeModel(model\_name="gemini-1.5-flash", generation\_config=generation\_config)*

    except Exception as e:

        print(f"Error initializing the model: {e}")

        return None

def generate\_book\_idea(model, genre, topic):

    prompt = f"""

    Generate a creative book idea in the genre '{genre}' about the topic '{topic}'.

    Include:

    1. Brief Synopsis

    2. Plot

    3. Main Characters

    4. Theme of the story

    5. Possible Plot Twists

    6. Ending of the story

    7. Lesson or Moral of the story

    """

    try:

        response = model.generate\_content(prompt)

        if hasattr(response, 'text') and response.text:

            return response.text

        else:

            return "No content generated."

    except Exception as e:

        return f"An error occurred: {e}"

@app.route('/')

def index():

    return render\_template('index.html')

@app.route('/generate', methods=['POST'])

def generate():

    genre = request.form['genre']

    topic = request.form['topic']

    if not genre or not topic:

        return render\_template('index.html', error="Both genre and topic are required.")

    model = initialize\_model(API\_KEY)

    if model:

        book\_idea = generate\_book\_idea(model, genre, topic)

        # *Store the generated book idea and genre/topic in session or redirect to another page*

        return redirect(url\_for('result', book\_idea=book\_idea, genre=genre, topic=topic))

    else:

        return render\_template('index.html', error="Failed to initialize the model.")

@app.route('/result')

def result():

    book\_idea = request.args.get('book\_idea')

    genre = request.args.get('genre')

    topic = request.args.get('topic')

    return render\_template('result.html', book\_idea=book\_idea, genre=genre, topic=topic)

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)