Project 30: Restaurant Recommender

Description:
This chatbot recommends restaurants based on your cravings, location, budget, and vibe. Whether you're in the mood for sushi, brunch with friends, or a romantic dinner—this foodie assistant gives curated suggestions and ambiance tips too!

---

restaurant_recommender.py

```
import openai
import os
import gradio as gr

# Load the OpenAI API key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Function to generate restaurant recommendations
def recommend_restaurants(user_input):
    # System message tells the model to act like a foodie concierge
    messages = [
        {
            "role": "system",
            "content": (
                "You are a local foodie and restaurant concierge. Based on a u
                "suggest restaurants with a short description for each. Incluc
                "If the user doesn't specify a location, give general suggesti
            )
        },
        {
            "role": "user",
            "content": f"I'm looking for a place to eat: {user_input}"
        }
    ]

    try:
        # Call OpenAI API to get dining recommendations
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",  # GPT-4 works great too
            messages=messages
        )

        # Return the curated restaurant picks
        return response['choices'][0]['message']['content'].strip()

    except Exception as e:
        # Handle any errors during the API call
        return f"Error: {str(e)}"

# Gradio interface for restaurant recommendation bot
iface = gr.Interface(
    fn=recommend_restaurants,                               # Function to rur
    inputs=gr.Textbox(lines=2, placeholder="e.g. I'm in NYC, craving Korean BE
    outputs="text",                                          # Display restau
    title="🍴 Restaurant Recommender",                       # App title
    description=(
        "Tell me what you’re in the mood for and I’ll suggest a few great rest
        "Try: 'Vegan spots in LA under $20', 'Romantic Italian dinner in Chica
    )
)

# Launch the restaurant recommender
iface.launch()
```