Project 63: Budget Summary Generator

Description:
This budgeting tool takes your list of income and expenses and produces a clear summary, including total savings, expense breakdown, and helpful tips. It's like having a friendly financial advisor who speaks human, not spreadsheets.

`budget_summary_generator.py`

```python
import openai
import os
import gradio as gr

# Load your OpenAI API Key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Function to generate budget summary
def generate_budget_summary(income, expenses):
    # Prompt to behave like a financial assistant
    messages = [
        {
            "role": "system",
            "content": (
                "You are a helpful budget assistant. Based on the user's incom
                "- Total income and total expenses\n"
                "- Surplus or deficit\n"
                "- Percentage of spending per category\n"
                "- A short financial insight or savings tip\n"
                "Keep the tone friendly and easy to understand."
            )
        },
        {
            "role": "user",
            "content": f"Income: {income}\nExpenses: {expenses}"
        }
    ]

    try:
        # Call OpenAI API
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",  # GPT-4 optional for more insight
            messages=messages
        )

        # Return the summary
        return response["choices"][0]["message"]["content"].strip()

    except Exception as e:
        return f"Error: {str(e)}"

# Gradio interface for budget summary
iface = gr.Interface(
    fn=generate_budget_summary,
    inputs=[
        gr.Textbox(label="Enter Income (e.g. Salary: 5000, Freelance: 1000)"),
        gr.Textbox(label="Enter Expenses (e.g. Rent: 1500, Food: 600, Internet
    ],
    outputs="text",
    title="💸 Budget Summary Generator",
    description="Enter your income and expenses to get a clear summary, spendi
)

# Launch the tool
iface.launch()
```