EduTutor AI: Personalized Learning with Generative AI

This project demonstrates how to build a simple personalized AI tutoring system using Flask (Python backend) and OpenAI's GPT model for generating tailored content based on topic and student level.

```
# app.py - Flask Backend
from flask import Flask, request, jsonify
import openai

app = Flask(__name__)
openai.api_key = "YOUR_OPENAI_API_KEY"

def generate_tutor_response(topic, level):
    prompt = f"Explain the topic '{topic}' to a student at the '{level}' level. Include
examples."
    response = openai.ChatCompletion.create(
        model="gpt-4",
        messages=[
            {"role": "system", "content": "You are a helpful AI tutor."},
            {"role": "user", "content": prompt}
        ]
    )
    return response['choices'][0]['message']['content']

@app.route("/learn", methods=["POST"])
def learn():
    data = request.get_json()
    topic = data.get("topic")
    level = data.get("level")
    if not topic or not level:
        return jsonify({"error": "Missing topic or level"}), 400
    result = generate_tutor_response(topic, level)
    return jsonify({"response": result})

if __name__ == "__main__":
    app.run(debug=True)
```

```html
<!-- frontend.html -->
<!DOCTYPE html>
<html>
<head>
  <title>EduTutor AI</title>
</head>
<body>
  <h1>EduTutor AI: Personalized Learning</h1>
  <label>Enter Topic:</label><br>
  <input type="text" id="topic"><br>
  <label>Select Level:</label><br>
  <select id="level">
    <option value="beginner">Beginner</option>
    <option value="intermediate">Intermediate</option>
    <option value="advanced">Advanced</option>
  </select><br><br>
  <button onclick="getExplanation()">Get Explanation</button>
  <p id="result"></p>
  <script>
    async function getExplanation() {
      const topic = document.getElementById('topic').value;
      const level = document.getElementById('level').value;
      const response = await fetch('/learn', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({ topic, level })
      });
      const data = await response.json();
      document.getElementById('result').innerText = data.response || data.error;
    }
  </script>
</body>
</html>
```