PROJECT INDUS

The code we've provided is a Python script for creating a basic web application using Streamlit. The primary purpose of this application is to perform text translation using the OpenAI GPT-3.5 Turbo model.
Here's a detailed description of the code development:

install packages : langchain, streamlit, openai
need openai api key
to run in terminal : "streamlit run indus.py"

1. Importing Libraries: The script begins by importing the necessary Python libraries and modules. These include:
- **`streamlit`:** Streamlit is used to create the web application and its user interface.
- **`langchain`:** Like a custom module for language-related functionalities.
- **`openai`:** The OpenAI Python library is used to interact with the GPT-3.5 Turbo model.

2. API Key Configuration: The OpenAI API key is set in the script to authenticate with the OpenAI service. It's important to keep API keys secure and not expose them directly in the code.

3. Translation Function: The `translate` function is defined to translate text using the GPT-3.5 Turbo model. It takes three parameters:
- **`text_to_translate`:** The text that needs to be translated.
- **`provider`:** The translation provider (Google, Bing, or OpenAI).
- **`target_language`:** The target language for translation.

Inside this function, a conversation with the GPT-3 model is set up. It includes a system message to instruct the model and a user message with the text to be translated. The model then generates a translation response based on this conversation.

4. Main Function: The `main` function is the core of the web application. Here's what it does:

- It sets the title of the Streamlit application to "Translation App."
- It provides an input field for the user to enter text they want to translate.
- It offers a selection box for the user to choose the translation provider (Google, Bing, or OpenAI).
- It provides another selection box for choosing the target language for translation.
- It initialises a variable `translated_text` to store the translated text.

The function then contains two conditional statements:
- The first conditional statement gets execute when the "Translate" button is clicked. It calls the `translate` function to perform the translation and displays the translated text on the web page.
- The second conditional statement gets execute when the "Translate New Text" button is clicked. It appears to be designed to keep track of the conversation history, but there are

some indentation issues with the code provided. The conversation history should be properly stored in `st.session_state.conversation` and displayed on the web page.

5. Running the Application: The script includes an `if __name__ == "__main__"` block to run the `main` function when the script is executed. This allows us to start the Streamlit web application.