

Setup:

- Create an environment
- Activate your environment
- Setup your openai [API key](#)
- Install Dependencies:
 - OpenAI
 - Langchain
 - Streamlit

Implementation:

The implementation process was initiated through a defining function that contains the template of the expecting generation format, which used to design the prompt.

The retrieved prompt is sent to a LLMchain where the prompt will be executed via the selected OpenAI model, “gpt-3.5-turbo” was selected to perform this task. The given prompt triggers the model to generate data, which comes in a text format. The returned data consists of both questions and answers in a single string.

A function that separates the questions from the answers (split_questions_answers) in two strings has been created. Another function that converts the resulting strings to lists has been created as well (convert_tolist).

Regarding the interface stage, Streamlit app framework used. The user can enter the topics and the desired number of questions through Streamlit input feature. After submitting the solutions for the displayed questions, a score is shown along with an option that allows the user to view the correct answers for each question.

User guide:

1. Initially, the user is asked to insert the topic they desire to be questioned about.
2. The user is then asked to insert the number of questions (minimum 1 and maximum is 10).
3. Questions related to the topic the user inserted in step one, and equal to the number they inserted in step 2 will be displayed. The user is then asked to insert the correct answer for each question.
4. Upon submission, the user will see their final score for the quiz.