**Task 3- Overview**

This task involves integrating OpenAI API calls into **mcq.py** to generate 5 challenging multiple-choice questions on topic chosen in dropdown, complexity level from radio buttons and submit the questions and generate the feedback for all questions.

**Task List**

1. **Understand the Boilerplate Code**: Review the structure and logic for multiple-choice question (MCQ) generation and evaluation.
   * Explore the data models and flow of JSON requests and responses.
2. **Implement Prompt Formatting**: Write prompts for generating MCQs based on a topic and complexity level.
3. **Integrate OpenAI API**: Complete the OpenAI API integration to dynamically generate MCQs and evaluate user submissions.
4. **Test and Debug**: Validate the functionality of each endpoint, ensuring correct processing of JSON data.

**Task**

Update boilerplate **mcq.py** code with two endpoints:

**Challenge 1: (**/mcq/generate)

1. Create a prompt, **json\_schema** and invoke OpenAI API to get the questions in following format. Output should be structured in json format.

{

    "Id": "Q1",

    "Question": "What is the capital of France?",

    "Options": [

        {

            "OptionIndex": 0,

            "OptionValue": "Berlin"

        },

        {

            "OptionIndex": 1,

            "OptionValue": "Madrid"

        },

        {

            "OptionIndex": 2,

            "OptionValue": "Paris"

        },

        {

            "OptionIndex": 3,

            "OptionValue": "Rome"

        }

    ],

    "CorrectOptionIndex": 2,

    "Complexity": "Basic"

}

**Challenge 2:**

**Update the application to use Python classes instead of Schema**