**Task 2- Overview**

This task involves integrating OpenAI API calls into **descriptive.py** to generate 3 challenging questions on topic chosen in dropdown and submit the questions and generate the feedback for all questions.

**Task List**

1. **Understand the Boilerplate Code**: Review the pre-existing structure and endpoints.
   * Understand the JSON response flow.
2. **Implement Prompt Formatting**: Write logic to create effective prompts for OpenAI API calls.
3. **Integrate OpenAI API**: Add OpenAI API integration for generating and evaluating responses.
4. **Test and Validate**: Ensure endpoints function correctly and validate all JSON responses.

**Task Description**

Update boilerplate **descriptive.py:**

**Challenge 1: (/generate-questions**)

Write a prompt to be generate following o/p

1. Generate 3 Question with properties "Id", "Question" and "ExpectedAnswer"

2. Json o/p

3. Answer in 2 or 3 sentences.

4. **DO** **NOT USE STRUCTURED OUTPUT** feature of OpenAI. Use JSON structure as below in prompt

[

  {

    "Id": 1,

    "Question": \_\_\_\_,

    "ExpectedAnswer": \_\_\_\_"

  }

]

1. Call OpenAI API to execute the prompt. Choose an appropriate System Message. Temperature=0.2
2. Debug and find the response text and remove all the special characters (if any) to ensure that has only JSON object.

**Important Note: In python use "{{" and "}}" if "{" and "}" are the characters needed in formatted string which begins with f" . . . . . "**

**Challenge 2 (/submitdescriptivequestions)**

1. Write a prompt instructing to evaluate all the questions in variable **user\_message**
   * You will need to debug and find the format of data in user\_message variable.
   * Get feedback for each question.
   * Overall rating should be between 1 and 10
2. Call OpenAI API to get response

**Tips for Success**

* Always test your prompts using simple examples.
* Use tools like Postman to test your API endpoints.