**Assignment 1 - Node Modules, Express, MongoDB and REST API**

#### Instructions

In this assignment you will continue the exploration of Node modules, Express, MongoDB and the REST API.

**Assignment Overview**

At the end of this assignment, you should have completed the following tasks to update the server:

* Applied MVC pattern
* Created a Node module using Express router to support the routes for the dishes REST API.
* Created a Node module using Express router to support the routes for quizzes and questions REST API.
* Created mongoose model – Fake with Array
* Connected with MongoDB

**Assignment Requirements**

+ A simple quiz is an interactive application designed to test users' knowledge on a particular topic or subject. It typically consists of a series of questions with multiple-choice options. Users select their answers and at the end of the quiz, the app displays the user's score and often offers a review of the questions and their correct answers.

+ Build your backend project that manages quiz and questions for a simple quiz app. The app allows users to implement the REST API (GET, POST, PUT, DELETE) endpoints /quizzes, / quizzes/:quizId and /question, /question/:questionId to interact with the MongoDB database namly SimpleQuiz.

+ Student can write the code based on the specific tasks

**1.** Create the schemas basing on the all requirements as below

Quiz: Represents a quiz or a test.

* Object\_id: Unique identifier for the quiz.
* title: Title or name of the quiz.
* description: Description or instructions for the quiz.
* questions: An array of question IDs associated with the quiz.

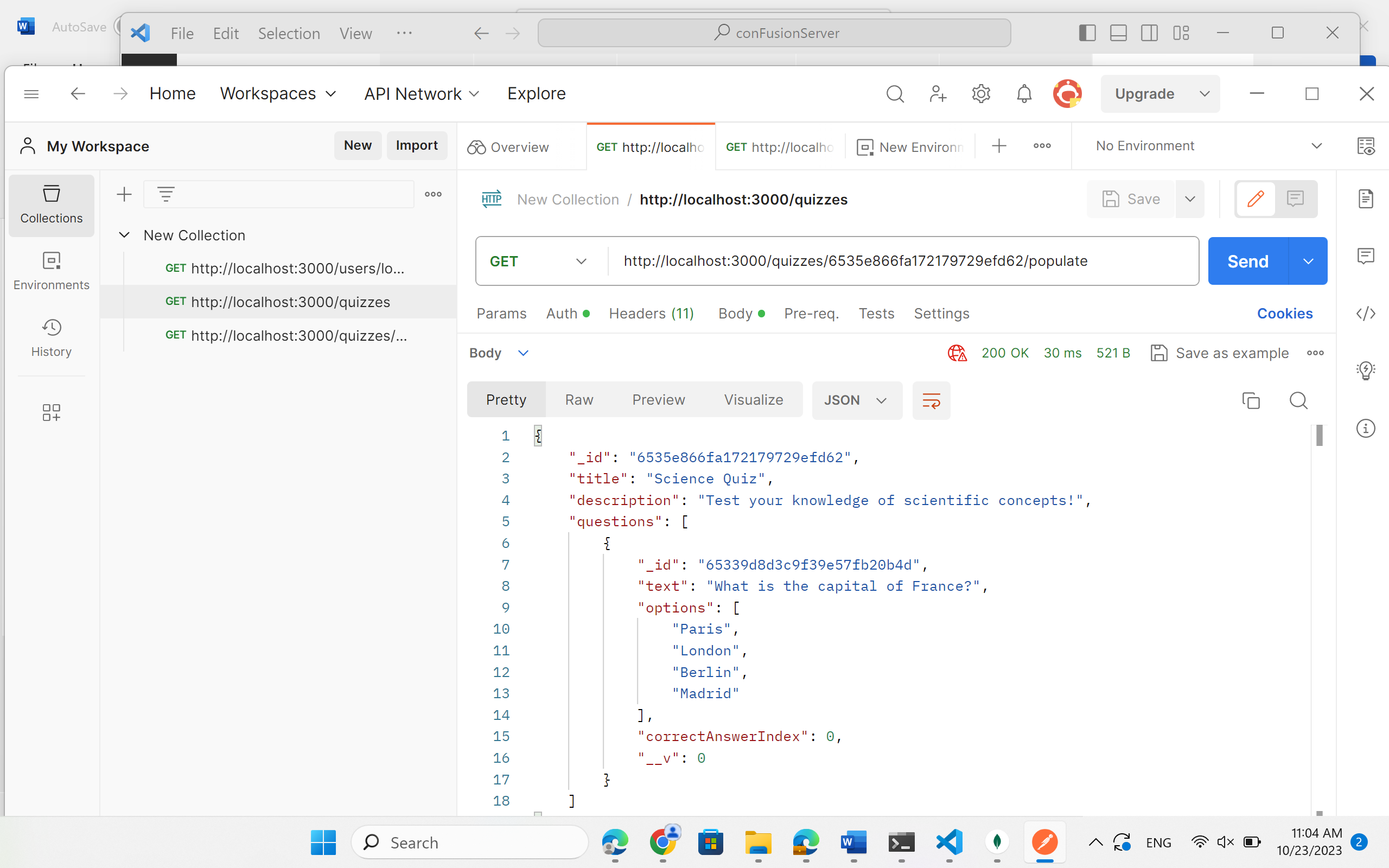
**Question: Represents a single question in the quiz.**

* Object\_id: Unique identifier for the question.
* text: The text of the question.
* options: An array of answer options for the question.
* keywords: An array of keywords for the question.
* correctAnswerIndex: Index of the correct answer within the options array.

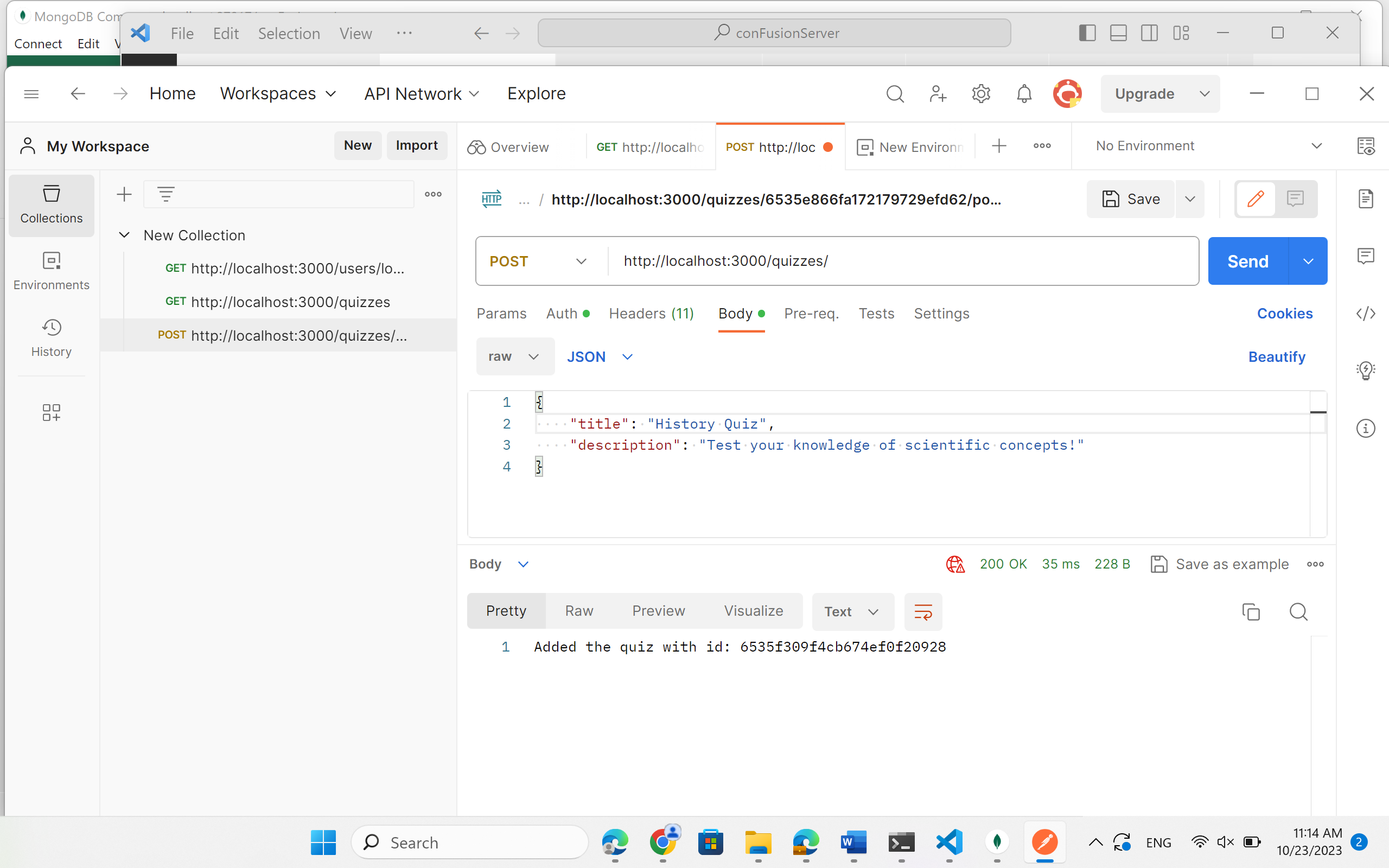
**2. Do the REST API (GET, POST, PUT, DELETE) endpoints /quizzes, /quizzes/:quizId and /question, /question/:questionId are implemented to interact with the MongoDB database namely SimpleQuiz**

Note: using Mongoose populate in order to display all questions in quiz when do endpoint /quizzes

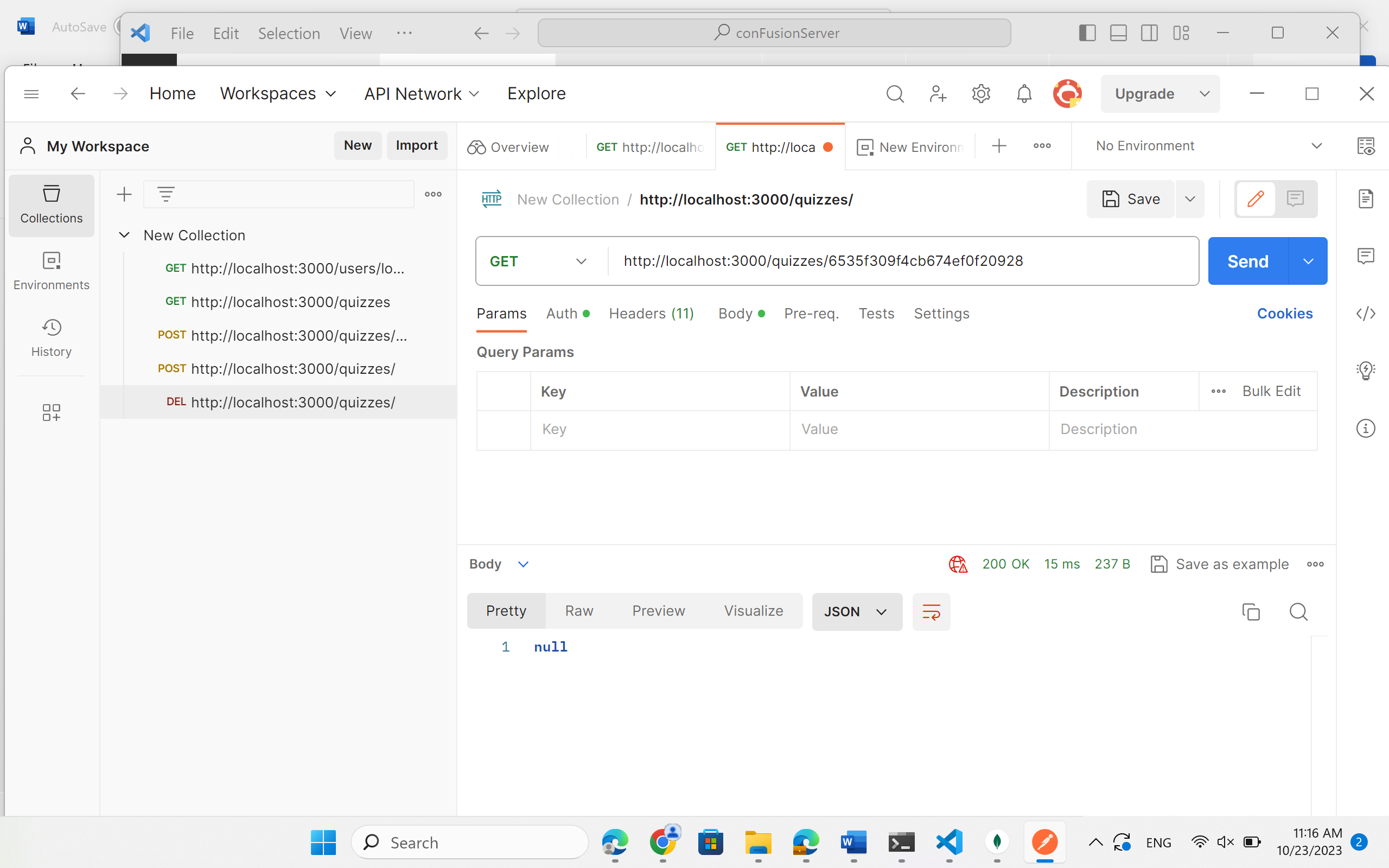
route GET/quizzes



Run chạy route POST/questions



Run chạy route DELETE/quizzes/:id



**3. Do the REST API (GET) endpoints /quizzes/:quizId/populate to match all questions with the word "capital"**

