Completing the 2 Microservices

Implementing Remaining Functionality

So far, we've implemented the createQuiz functionality in our Quiz Service. Now we need to complete two additional core functions:

- Getting quiz questions (getQuizQuestions)
- Submitting quiz answers and calculating results (calculateResult)

Both of these functions will also require communication with the Question Service through our Feign client.

Retrieving Quiz Questions

When a user wants to take a quiz, we need to fetch all the questions for that quiz. Here's how we implement this in the Quiz Service:

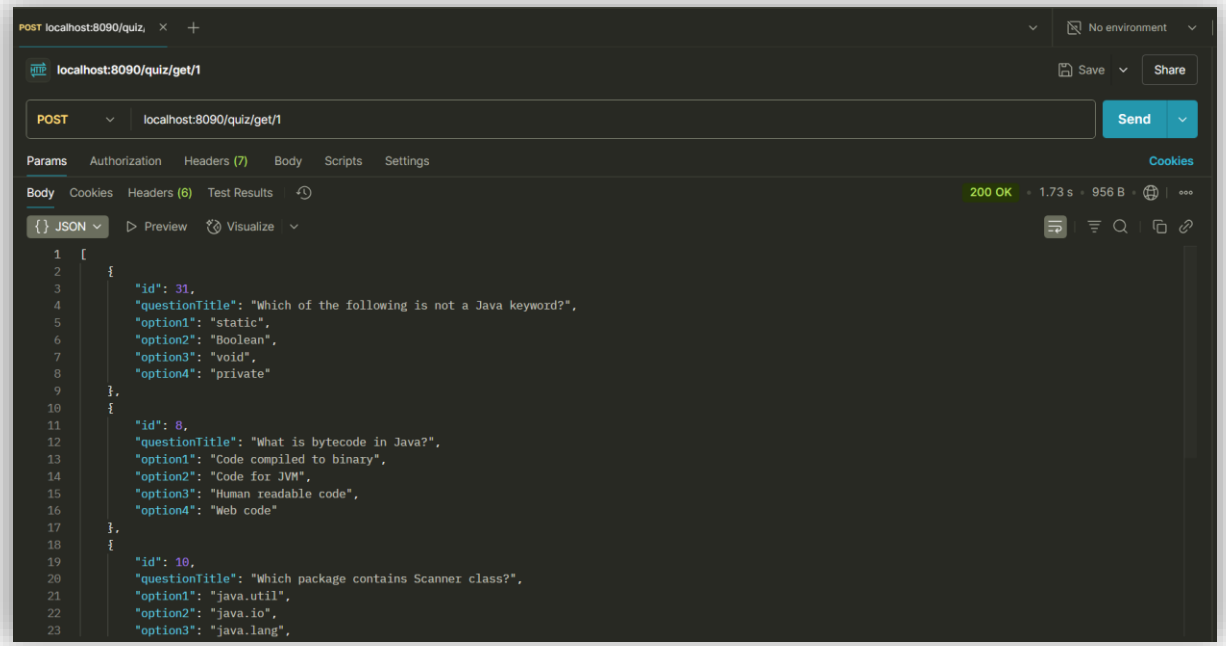
```
public ResponseEntity<List<QuestionWrapper>> getQuizQuestions(Integer id) {
    // Find the quiz by ID from the database
    Quiz quiz = quizDao.findById(id).get();

    // Extract the list of question IDs stored in the quiz
    List<Integer> questionIds = quiz.getQuestionIds();

    // Call the Question Service through Feign client to get the actual questions
    // This makes an HTTP request to the Question Service's getQuestionsFromId endpoint
    ResponseEntity<List<QuestionWrapper>> questions = quizInterface.getQuestionsFromId(questionIds);

    return questions;
}
```

- We retrieve the local data we have (quiz with question IDs)
- We then call another service to get related data (questions)
- We return the combined result to the client

The corresponding Feign client method in our QuizInterface is:

```
@PostMapping("question/getQuestions")
public ResponseEntity<List<QuestionWrapper>> getQuestionsFromId(@RequestBody List<Integer> questionIds);
```

Testing Question Retrieval



The response contains a list of questions with their options, but without the correct answers. This is appropriate for a quiz that a user is about to take.

Calculating Quiz Results

When a user submits their quiz answers, we need to calculate their score. Since the correct answers are stored in the Question Service, we need to forward the user's responses to get the score:

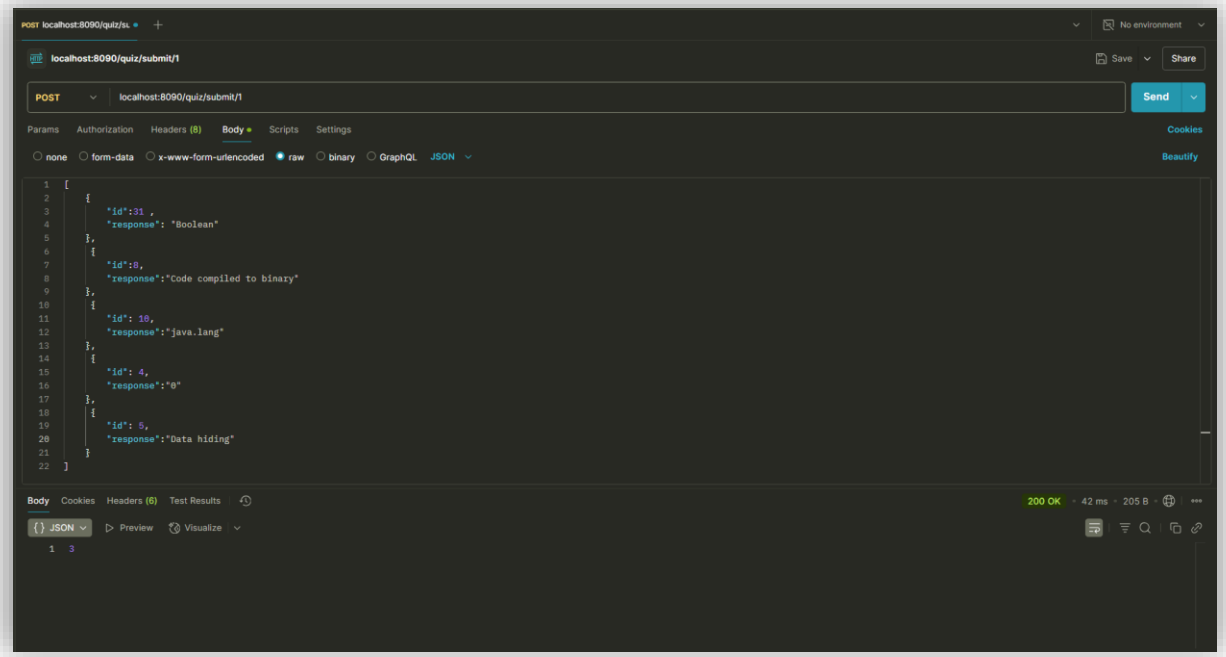
```java
public ResponseEntity<Integer> calculateResult(Integer id, List<Response> responses) {
    // Call the Question Service's getScore endpoint with the user's responses
    // The Question Service will compare the responses with the correct answers
    ResponseEntity<Integer> score = quizInterface.getScore(responses);

    return score;
}
```

- Takes a list of user responses (question ID and selected answer)
- Forwards these responses to the Question Service
- Returns the calculated score back to the client

The corresponding Feign client method is:

```java
@PostMapping("question/getScore")
public ResponseEntity<Integer> getScore(@RequestBody List<Response> responses);
```

Testing Score Calculation



The response contains the user's score, calculated by the Question Service.