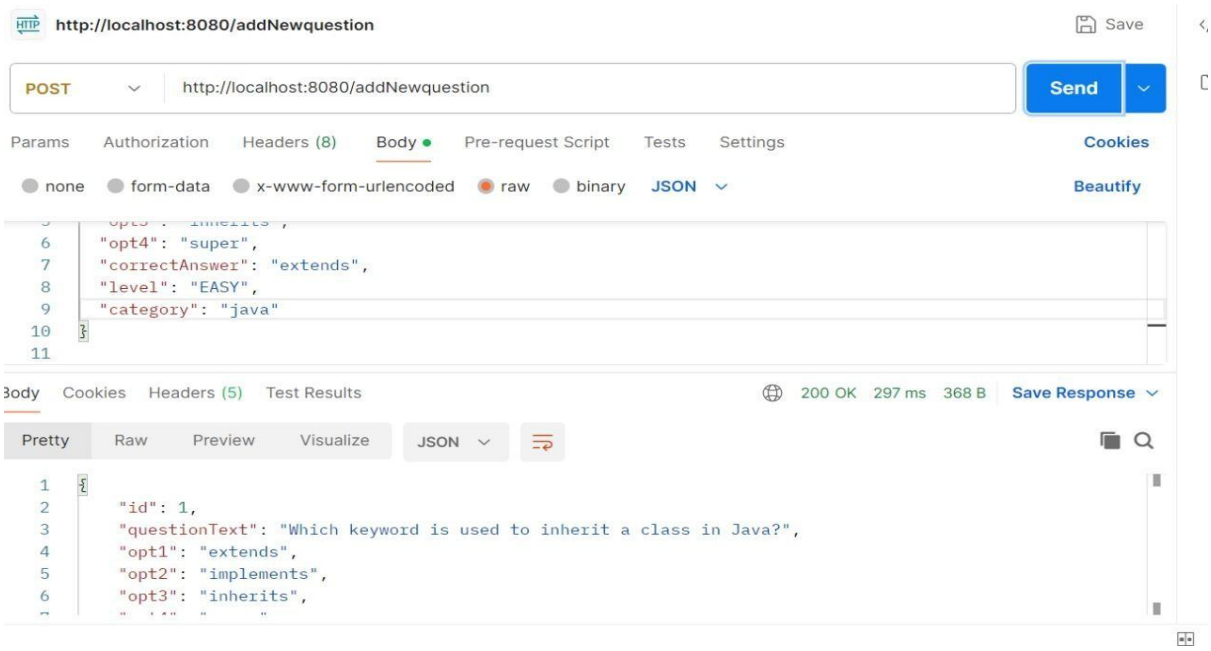
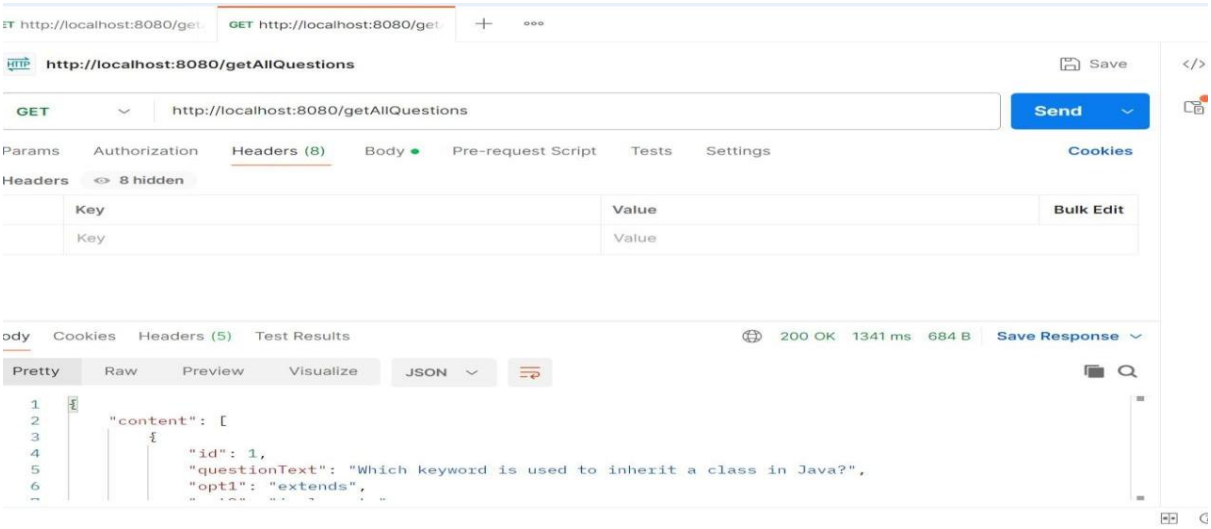
Quizz project:- Name:Praveen Jeennala

Implementing questions:-

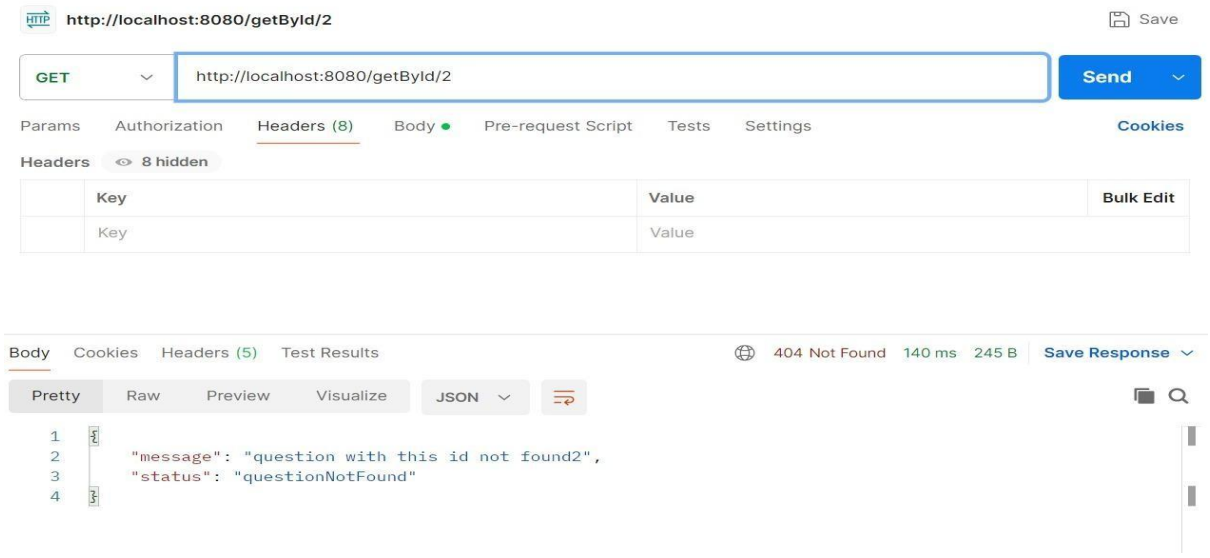
1.addNewQuestion:-



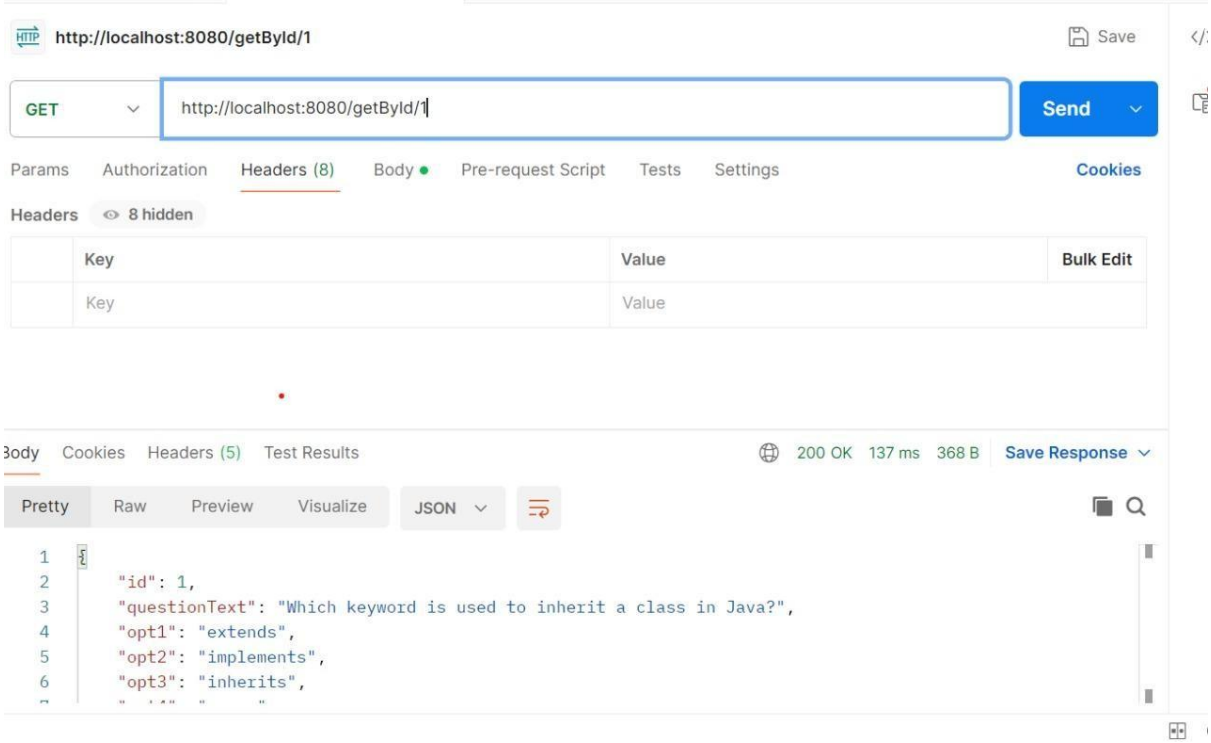
2.getAllQuestions:-



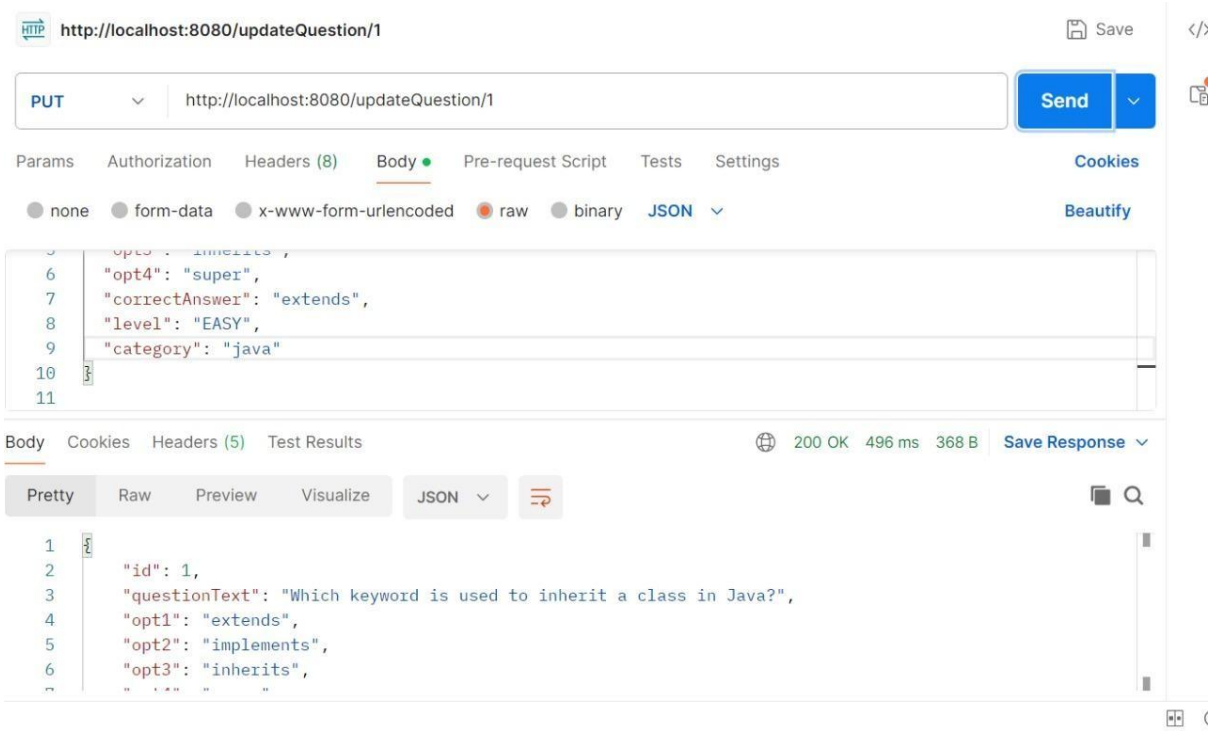
3.getById implementing with custom exception:--



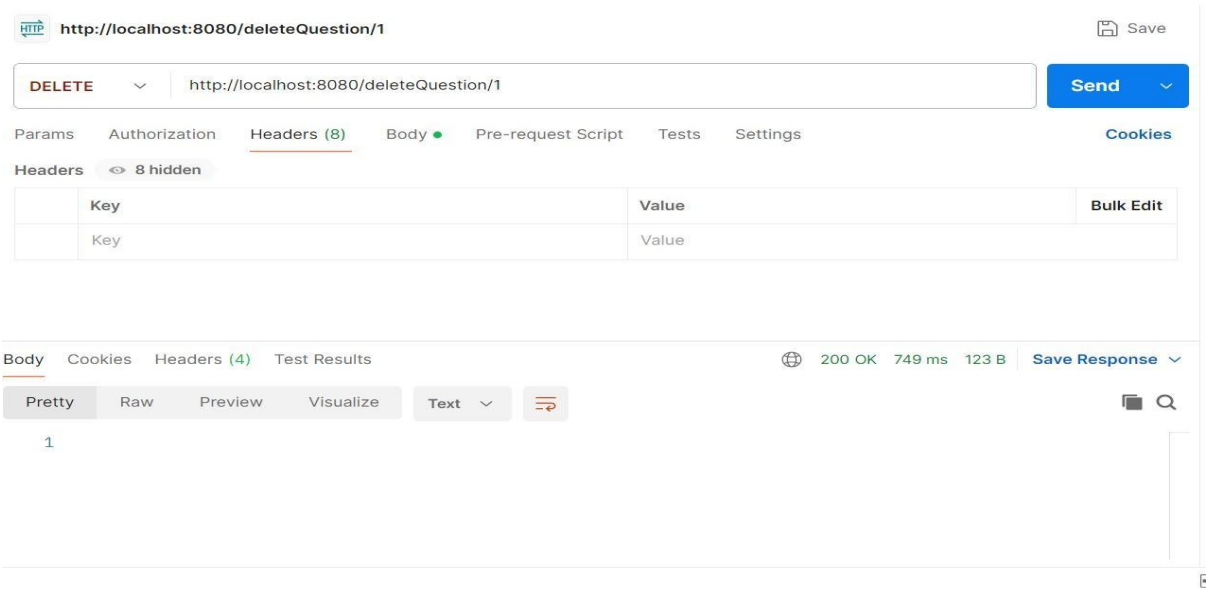
4.getById:;



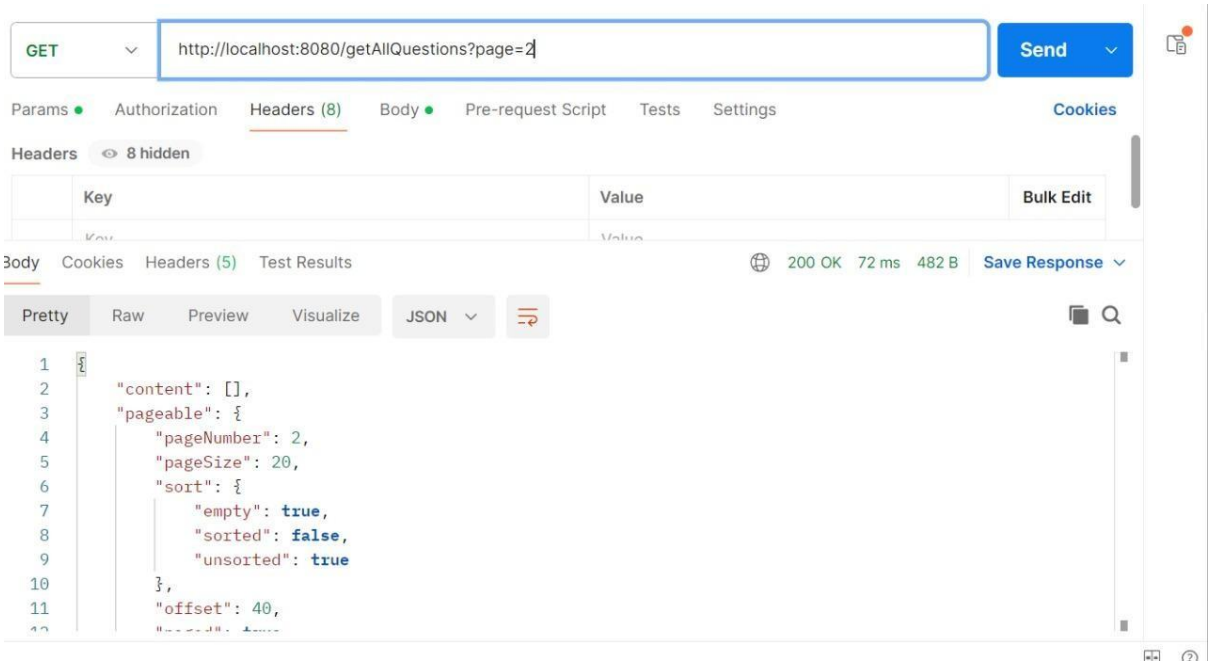
5.updateQuestion:-



6.deleteQuestion:-

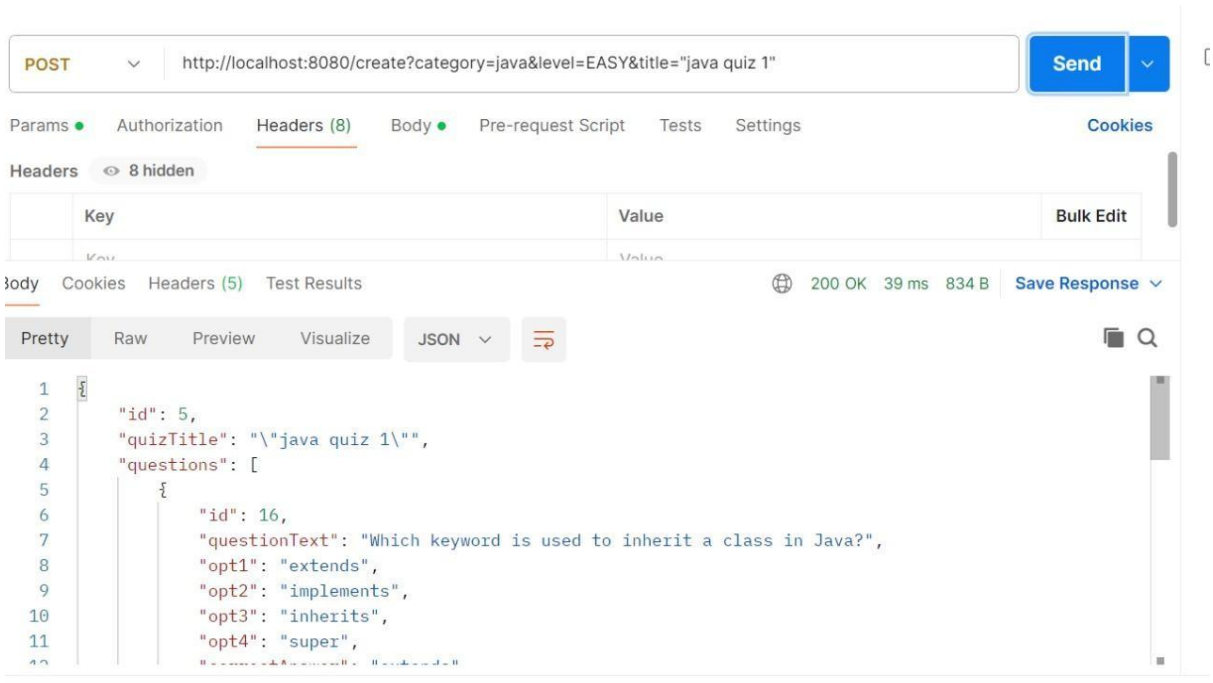


7.getAll question with implementing pagenation:--

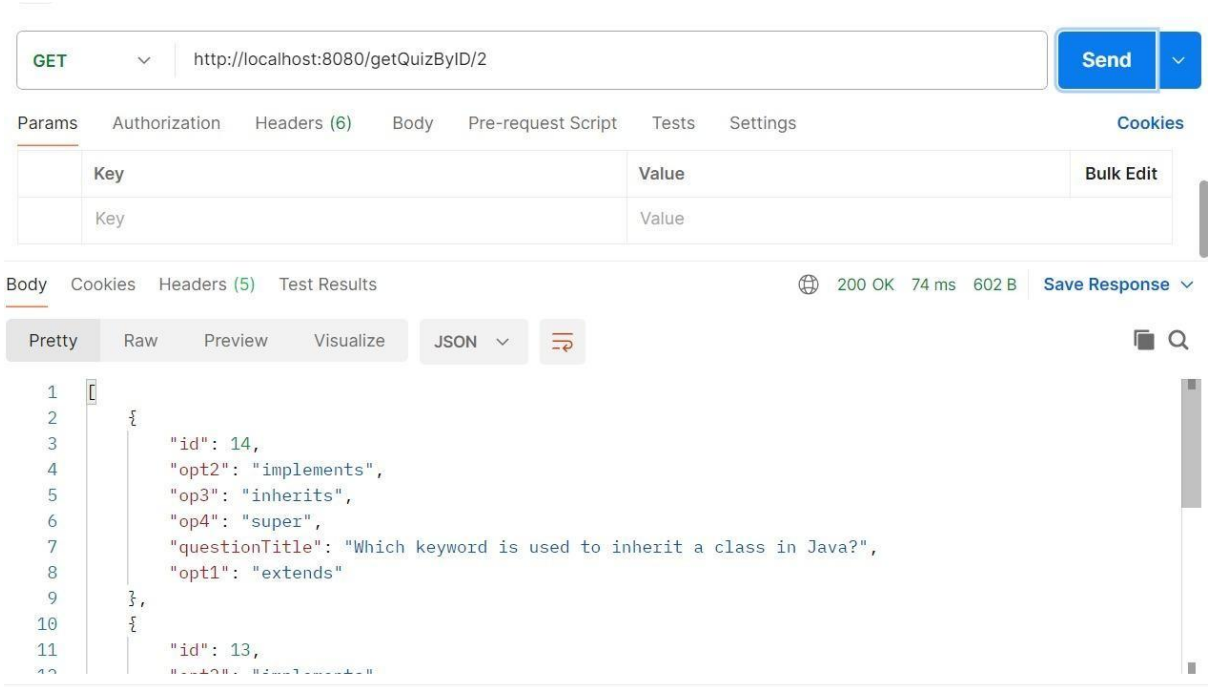


Quiz implementation

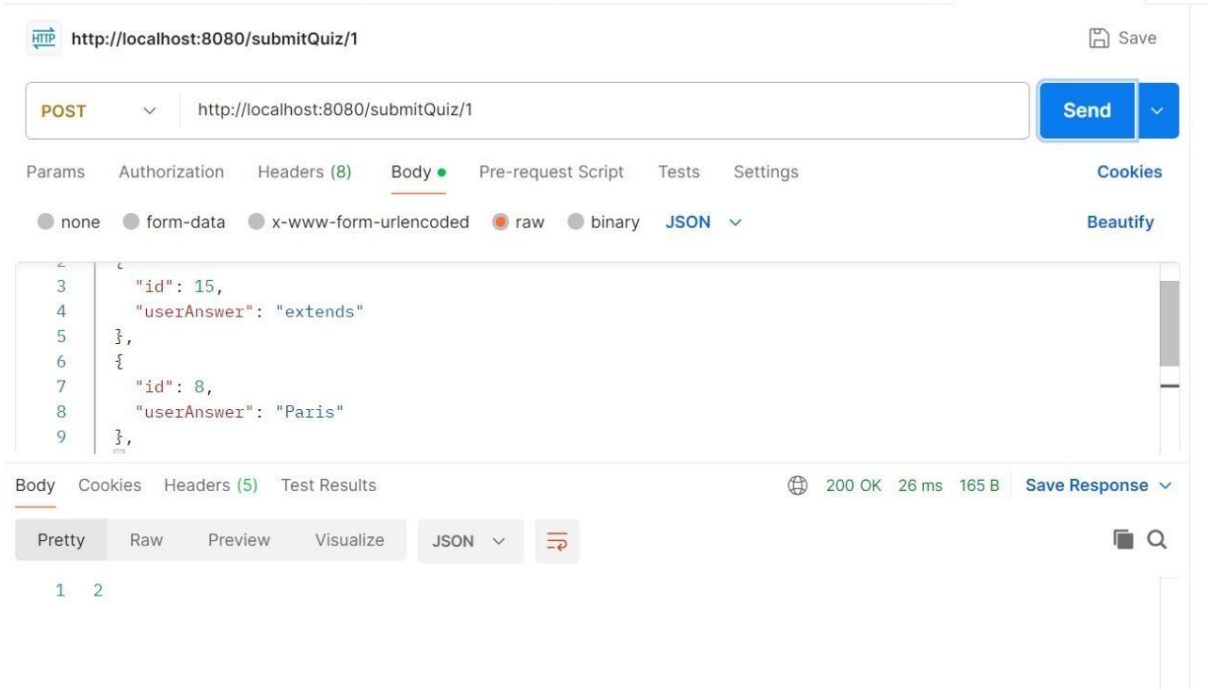
1.Create quiz:-



2.Get Quiz BY id:--



3.Submitting quiz with getting score on console:-



4.Implementing Aop:--

```java
package com.example.demo.aop;

import org.aspectj.lang.JoinPoint;
import org.aspectj.lang.annotation.Aspect;
import org.aspectj.lang.annotation.Before;
import org.springframework.stereotype.Component;

@Aspect
@Component
public class LoggingAspect {

    @Before("execution(* com.example.demo.service.*.*(..))")
    public void logBefore(JoinPoint joinPoint) {
        System.out.println("AOP BEFORE: Calling method - " + joinPoint.getSignature().g
    }
}
```

Output of Aop::-

```
2025-08-09T09:09:30.077+05:30  INFO 19392 --- [quizSpringBoot-App] [nio-8080-exec-1] o.s.web
2025-08-09T09:09:31.208+05:30  INFO 19392 --- [quizSpringBoot-App] [nio-8080-exec-8] o.sprin
AOP BEFORE: Calling method - createQuiz
Hibernate:
    SELECT
        *
    FROM
        question q
    WHERE
        q.category = ?
```

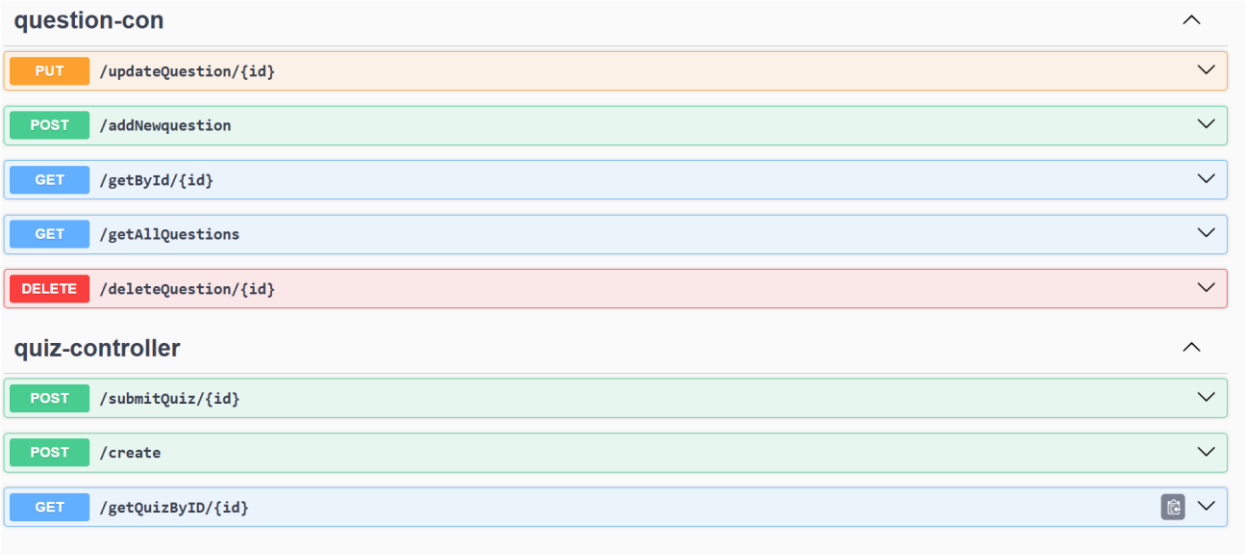
5.Implemented logging:-

```java
    @GetMapping("/getQuizByID/{id}")
    public List<QuestionWrapper> getQuizQuestions(@PathVariable Integer id){
     log.info("quiz starting....");

        return quizService.getQuizQuestions(id);
    }
```

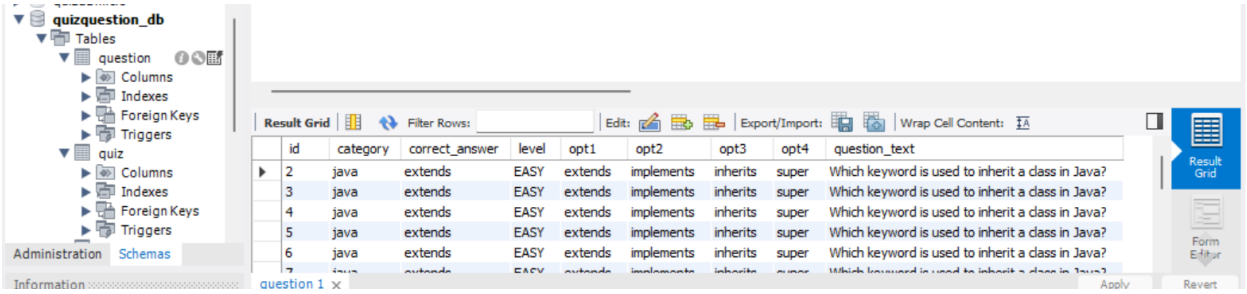
6.Implementing validations:-

```java
@Id
@GeneratedValue(strategy = GenerationType.IDENTITY)
private Long id;
@NotNull(message="question not be null")
private String questionText;

private String opt1;
```

7.Swagger ui:-



8.DataBase:-



9.docs

Pretty-print ☑


```
{
  "openapi": "3.1.0",
  "info": {
    "title": "OpenAPI definition",
    "version": "v0"
  },
  "servers": [
    {
      "url": "http://localhost:8080",
      "description": "Generated server url"
    }
  ],
  "paths": {
    "/updateQuestion/{id}": {
      "put": {
        "tags": [
          "question-con"
        ],
        "operationId": "updateQuestion",
        "parameters": [
          {
            "name": "id",
            "in": "path",
            "required": true,
            "schema": {
              "type": "integer",
              "format": "int64"
            }
          }
        ],
        "requestBody": {
          "content": {
            "application/json": {
              "schema": {
                "$ref": "#/components/schemas/Question"
```