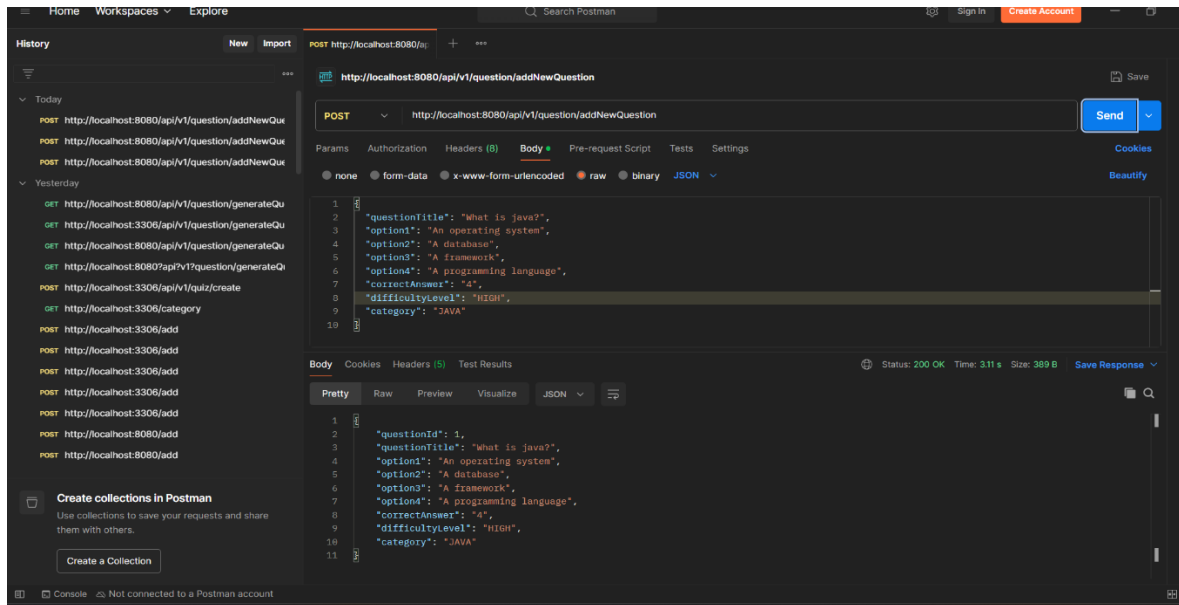
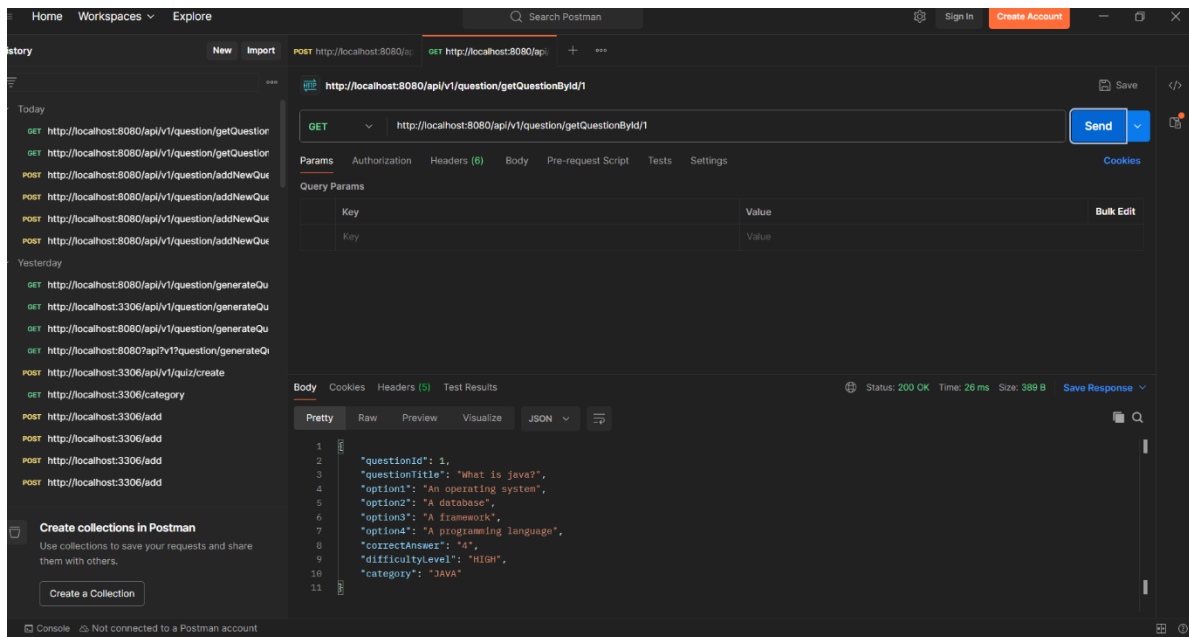


Quiz-App Mono repo

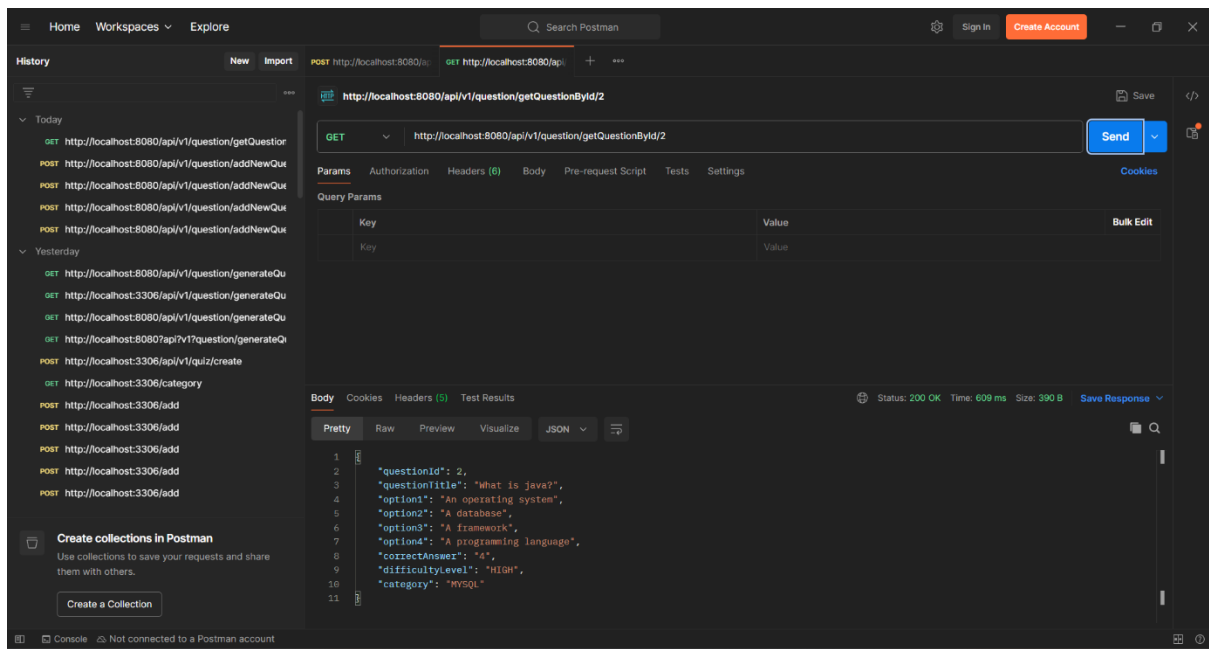
1 . Add New Question



2. Get Question By Id/1



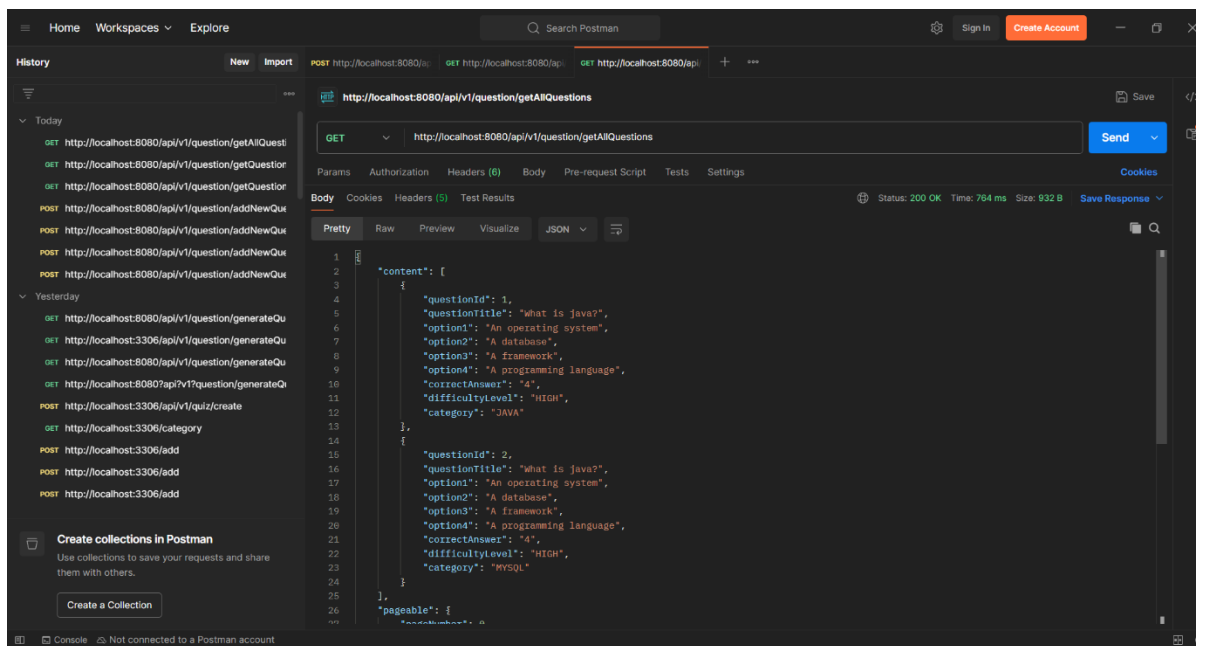
3. Get Question By Id/2



The screenshot shows the Postman interface with a GET request to `http://localhost:8080/api/v1/question/getQuestionById/2`. The response is a JSON object with the following structure:

```
1 {
2   "questionId": 2,
3   "questionTitle": "What is java?",
4   "option1": "An operating system",
5   "option2": "A database",
6   "option3": "A framework",
7   "option4": "A programming language",
8   "correctAnswer": "A",
9   "difficultyLevel": "HIGH",
10  "category": "MYSQL"
11 }
```

4. Get All Question



The screenshot shows the Postman interface with a GET request to `http://localhost:8080/api/v1/question/getAllQuestions`. The response is a JSON object with a `content` array and a `pageable` object:

```
1 {
2   "content": [
3     {
4       "questionId": 1,
5       "questionTitle": "What is java?",
6       "option1": "An operating system",
7       "option2": "A database",
8       "option3": "A framework",
9       "option4": "A programming language",
10      "correctAnswer": "A",
11      "difficultyLevel": "HIGH",
12      "category": "JAVA"
13     },
14     {
15       "questionId": 2,
16       "questionTitle": "What is java?",
17       "option1": "An operating system",
18       "option2": "A database",
19       "option3": "A framework",
20       "option4": "A programming language",
21       "correctAnswer": "A",
22       "difficultyLevel": "HIGH",
23       "category": "MYSQL"
24     }
25   ],
26   "pageable": {
27     "page": 0,
28     "size": 20,
29     "totalElements": 2,
30     "totalPages": 1,
31     "last": true,
32     "first": true,
33     "sort": "id",
34     "direction": "ASC"
35   }
36 }
```

History New Import POST http://localhost:8080/api/ GET http://localhost:8080/api/ GET http://localhost:8080/api/ GET http://localhost:8080/api/ POST http://localhost:8080/api/ + ...

Today

- GET http://localhost:8080/api/v1/question/getBYCatDiff?category=JAVA&difficultyLevel=HIGH
- GET http://localhost:8080/api/v1/question/getBYCatDiff?category=JAVA&difficultyLevel=HIGH
- GET http://localhost:8080/api/v1/question/getBYCatDiff?category=JAVA&difficultyLevel=HIGH
- GET http://localhost:8080/api/v1/question/getBYCatDiff?category=JAVA&difficultyLevel=HIGH
- GET http://localhost:8080/api/v1/question/http://localhost:8080/api/v1/question/getAllQuestions
- GET http://localhost:8080/api/v1/question/getAllQuestions
- GET http://localhost:8080/api/v1/question/getAllQuestions
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion

Yesterday

- GET http://localhost:8080/api/v1/question/generateQuiz
- GET http://localhost:3306/api/v1/question/generateQuiz
- GET http://localhost:8080/api/v1/question/generateQuiz
- GET http://localhost:8080/api/v1/question/generateQuiz

Create collections in Postman

Use collections to save your requests and share them with others.

Create a Collection

GET http://localhost:8080/api/v1/question/getBYCatDiff?category=JAVA&difficultyLevel=HIGH

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value
<input checked="" type="checkbox"/> category	JAVA
<input checked="" type="checkbox"/> difficultyLevel	HIGH

Body Cookies Headers (5) Test Results Status: 200 OK Time: 26 ms Size: 391 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "questionId": 1,
3   "questionTitle": "What is java?",
4   "option1": "An operating system",
5   "option2": "A database",
6   "option3": "A framework",
7   "option4": "A programming language",
8   "correctAnswer": "4",
9   "difficultyLevel": "HIGH",
10  "category": "JAVA"
11 }
12
13
```

Home Workspaces Explore Search Postman Sign in Create Account

History New Import POST http://localhost:8080/api/ GET http://localhost:8080/api/ GET http://localhost:8080/api/ GET http://localhost:8080/api/ POST http://localhost:8080/api/ + ...

Today

- GET http://localhost:8080/api/v1/question/getBYCatDiff?category=MYSQL&difficultyLevel=HIGH
- GET http://localhost:8080/api/v1/question/http://localhost:8080/api/v1/question/getAllQuestions
- GET http://localhost:8080/api/v1/question/getAllQuestions
- GET http://localhost:8080/api/v1/question/getAllQuestions
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion
- POST http://localhost:8080/api/v1/question/addNewQuestion

Yesterday

- GET http://localhost:8080/api/v1/question/generateQuiz
- GET http://localhost:3306/api/v1/question/generateQuiz
- GET http://localhost:8080/api/v1/question/generateQuiz
- GET http://localhost:8080/api/v1/question/generateQuiz
- POST http://localhost:3306/api/v1/quiz/create
- GET http://localhost:3306/category
- POST http://localhost:3306/add

Create collections in Postman

Use collections to save your requests and share them with others.

Create a Collection

GET http://localhost:8080/api/v1/question/getBYCatDiff?category=MYSQL&difficultyLevel=HIGH

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

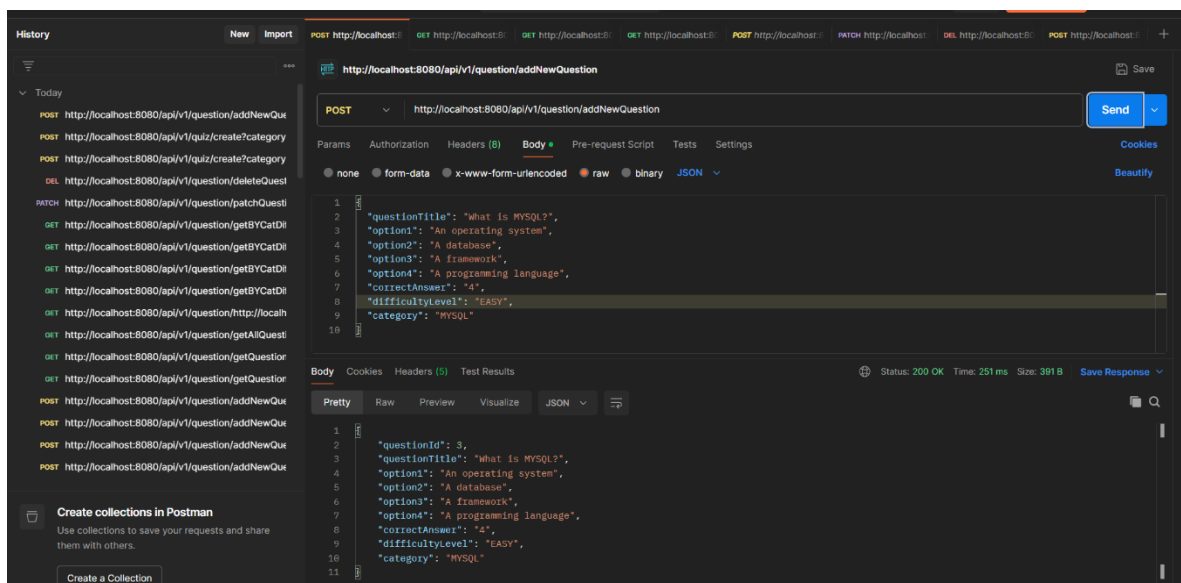
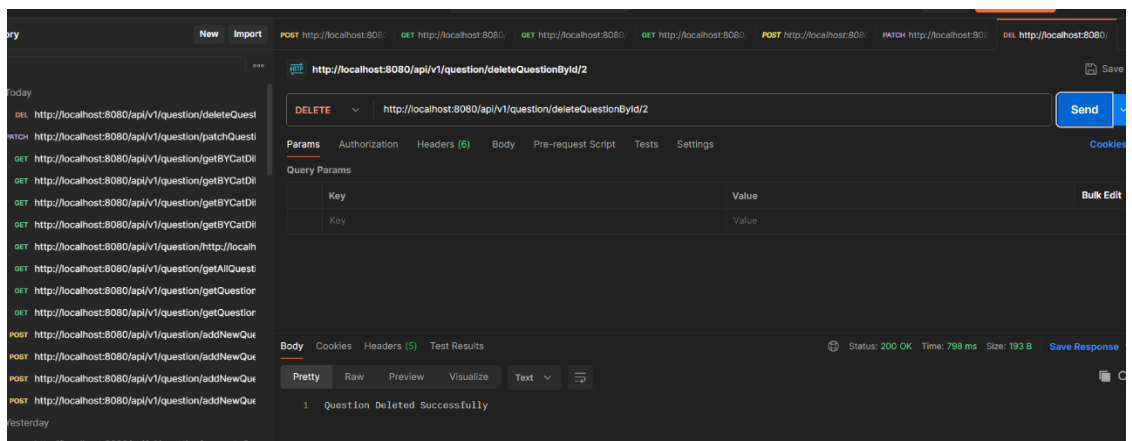
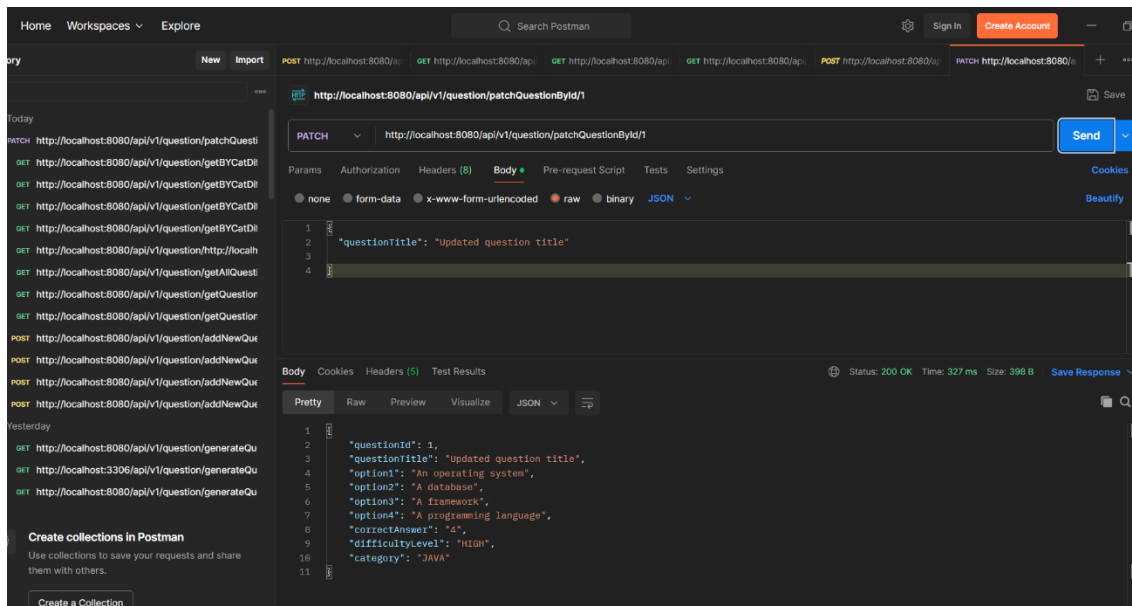
Query Params

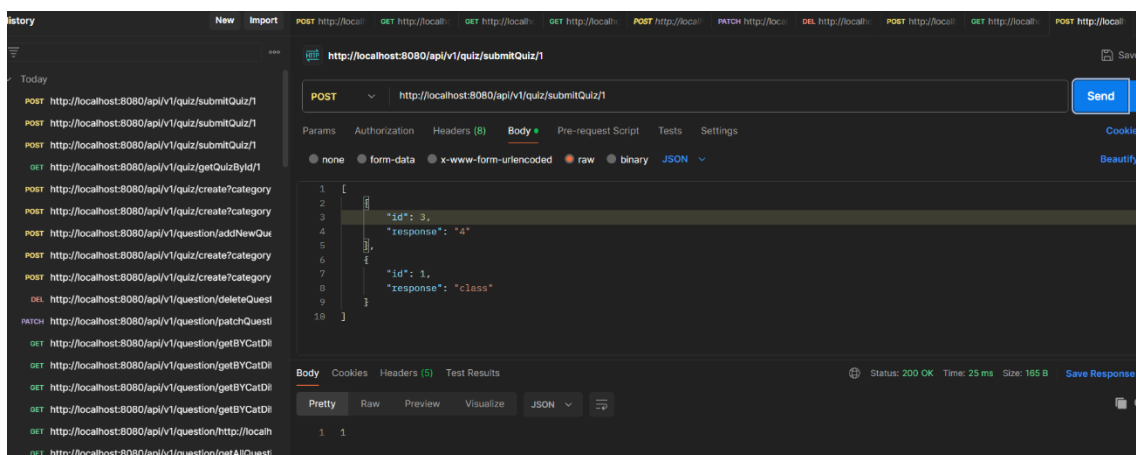
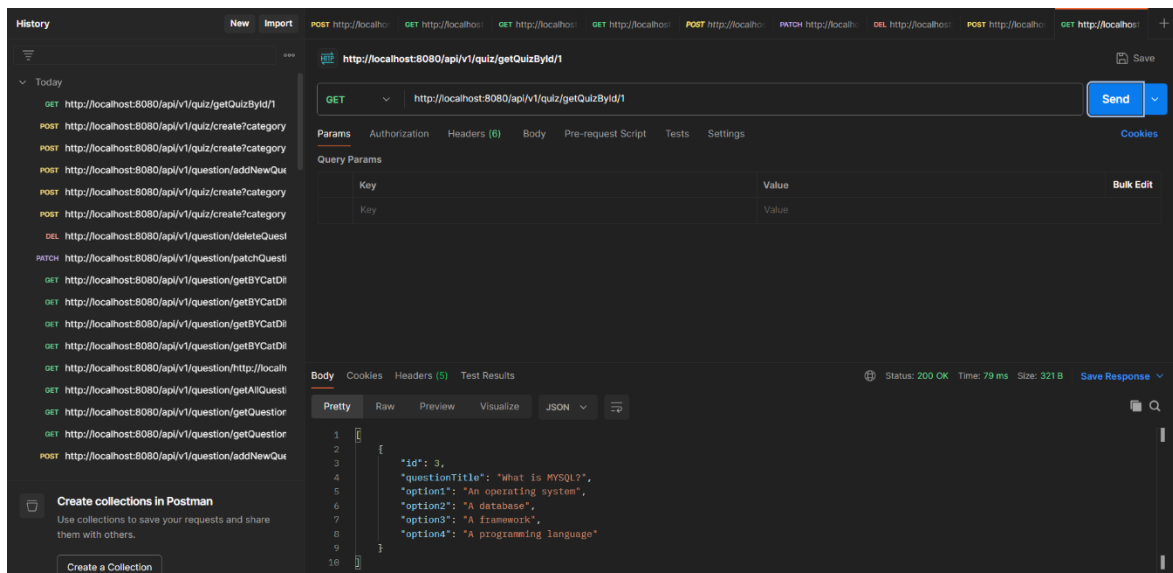
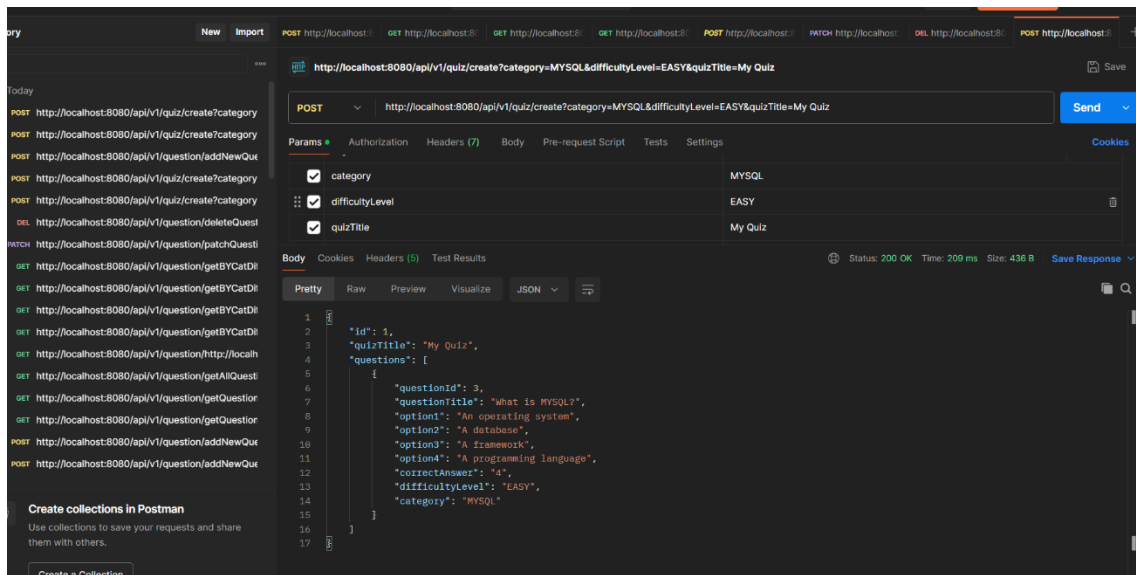
Key	Value
<input checked="" type="checkbox"/> category	MYSQL
<input checked="" type="checkbox"/> difficultyLevel	HIGH

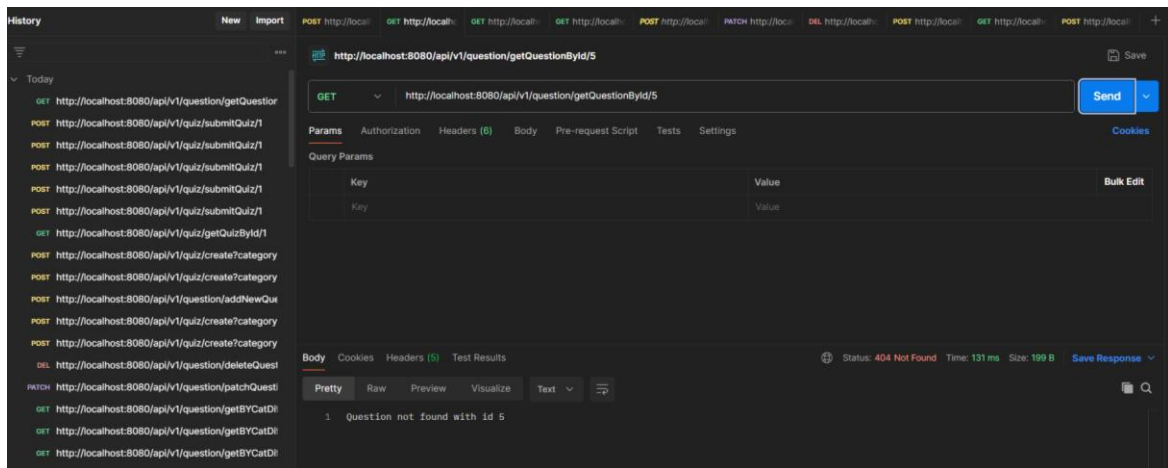
Body Cookies Headers (5) Test Results Status: 200 OK Time: 296 ms Size: 392 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "questionId": 2,
3   "questionTitle": "What is java?",
4   "option1": "An operating system",
5   "option2": "A database",
6   "option3": "A framework",
7   "option4": "A programming language",
8   "correctAnswer": "4",
9   "difficultyLevel": "HIGH",
10  "category": "MySQL"
11 }
12
13
```







OUTPUT:-

```

eclipse-workspace - quizapp_monorepo/src/main/resources/application.properties - Eclipse IDE
File Edit Navigate Search Project Run Window Help
QuizApp monorepo - QuizAppApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\java.exe (10-Aug-2025, 1:53:36pm elapsed:041:20) [pid: 14676]

:: Spring Boot :: (v3.3.4)

2025-08-10T14:24:59.414+05:30 INFO 14676 --- [quiz] [ restartedMain] c.wipro.jv.quizmono.QuizAppApplication : Starting QuizAppApplication
2025-08-10T14:24:59.415+05:30 INFO 14676 --- [quiz] [ restartedMain] c.wipro.jv.quizmono.QuizAppApplication : No active profile set, falli
2025-08-10T14:25:00.322+05:30 INFO 14676 --- [quiz] [ restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data J
2025-08-10T14:25:00.370+05:30 INFO 14676 --- [quiz] [ restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data reposi
2025-08-10T14:25:00.733+05:30 INFO 14676 --- [quiz] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port
2025-08-10T14:25:00.737+05:30 INFO 14676 --- [quiz] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-08-10T14:25:00.738+05:30 INFO 14676 --- [quiz] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Ap
2025-08-10T14:25:00.778+05:30 INFO 14676 --- [quiz] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedde
2025-08-10T14:25:00.778+05:30 INFO 14676 --- [quiz] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext:
2025-08-10T14:25:00.859+05:30 INFO 14676 --- [quiz] [ restartedMain] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing Persi
2025-08-10T14:25:00.867+05:30 INFO 14676 --- [quiz] [ restartedMain] o.h.c.internal.RegionFactoryInitiator : No LoadTimeWeaver setup: ign
2025-08-10T14:25:00.919+05:30 INFO 14676 --- [quiz] [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-2 - Starting...
2025-08-10T14:25:00.923+05:30 INFO 14676 --- [quiz] [ restartedMain] com.zaxxer.hikari.pool.HikariPool : HikariPool-2 - Added connect
2025-08-10T14:25:00.984+05:30 INFO 14676 --- [quiz] [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-2 - Start complet
2025-08-10T14:25:01.178+05:30 INFO 14676 --- [quiz] [ restartedMain] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000489: No JTA platform i
2025-08-10T14:25:01.358+05:30 INFO 14676 --- [quiz] [ restartedMain] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManag
2025-08-10T14:25:01.545+05:30 WARN 14676 --- [quiz] [ restartedMain] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is e
2025-08-10T14:25:01.768+05:30 INFO 14676 --- [quiz] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running
2025-08-10T14:25:01.795+05:30 INFO 14676 --- [quiz] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080
2025-08-10T14:25:01.802+05:30 INFO 14676 --- [quiz] [ restartedMain] c.wipro.jv.quizmono.QuizAppApplication : Started QuizAppApplication :

```

```
eclipse-workspace - quizapp_monorepo/src/main/resources/application.properties - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Declaration Progress Console X
quizapp_monorepo - QuizAppApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (10-Aug-2025, 1:53:36pm elapsed: 0:41:23) [pid: 14676]
Hibernate:
select
    q1_0.question_id,
    q1_0.category,
    q1_0.correct_answer,
    q1_0.difficulty_level,
    q1_0.option1,
    q1_0.option2,
    q1_0.option3,
    q1_0.option4,
    q1_0.question_title
from
    question q1_0
where
    q1_0.category=?
    and q1_0.difficulty_level=?
Hibernate:
insert
into
    quiz
    (quiz_title)
values
    (?)
Hibernate:
insert
into
    quiz_questions
    (quiz_id, questions_question_id)
values
    (?, ?)
```

```
eclipse-workspace - quizapp_monorepo/src/main/resources/application.properties - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Declaration Progress Console X
quizapp_monorepo - QuizAppApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (10-Aug-2025, 1:53:36pm elapsed: 0:41:32) [pid: 14676]
values
    (?)
Hibernate:
insert
into
    quiz_questions
    (quiz_id, questions_question_id)
values
    (?, ?)
Hibernate:
select
    q1_0.id,
    q1_0.quiz_title
from
    quiz q1_0
where
    q1_0.id=?
Hibernate:
select
    q1_0.quiz_id,
    q1_1.question_id,
    q1_1.category,
    q1_1.correct_answer,
    q1_1.difficulty_level,
    q1_1.option1,
    q1_1.option2,
    q1_1.option3,
    q1_1.option4,
    q1_1.question_title
from
```


Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- one_2_one_mapping
- question_service_db
 - question
 - quiz
 - quiz_questions
 - quiz_service_db
 - quizserver
 - quizserver_db
 - question
 - quiz
 - quiz_questions
 - Views
 - Stored Procedures
 - Functions
- sakila
- student
- sys
- world

Administration Schemas

Information

Schema: question_service_db

SQL File 10* SQL File 11* question quiz SQL File 4* question question question

1 • SELECT * FROM quizserver_db.question;

Result Grid

question_id	category	correct_answer	difficulty_level	option1	option2	option3	option4	question_title
1	JAVA	4	HIGH	An operating system	A database	A framework	A programming language	Updated question title
3	MYSQL	4	EASY	An operating system	A database	A framework	A programming language	What is MySQL?

question 1 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
2	10:07:45	SELECT * FROM quizservice.quiz LIMIT 0, 1000	0 row(s) returned	0.015 sec / 0.000 sec
3	13:24:16	create database quizserver_db	1 row(s) affected	0.078 sec
4	13:25:45	SELECT * FROM one_2_one_mapping.table LIMIT 0, 1000	3 row(s) returned	0.031 sec / 0.000 sec
5	13:25:59	SELECT * FROM question_service_db.question LIMIT 0, 1000	0 row(s) returned	0.078 sec / 0.000 sec
6	15:08:33	SELECT * FROM quizservice.question LIMIT 0, 1000	0 row(s) returned	0.032 sec / 0.000 sec
7	15:09:18	SELECT * FROM quizserver_db.question LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- one_2_one_mapping
- question_service_db
 - question
 - quiz
 - quiz_questions
 - quiz_service_db
 - quizserver
 - quizserver_db
 - question
 - quiz
 - quiz_questions
 - Views
 - Stored Procedures
 - Functions
- sakila
- student
- sys
- world

Administration Schemas

Information

Schema: question_service_db

SQL File 10* SQL File 11* question quiz SQL File 4* question question question quiz

1 • SELECT * FROM quizserver_db.quiz;

Result Grid

id	quiz_title
1	My Quiz

quiz 1 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
3	13:24:16	create database quizserver_db	1 row(s) affected	0.078 sec
4	13:25:45	SELECT * FROM one_2_one_mapping.table LIMIT 0, 1000	3 row(s) returned	0.031 sec / 0.000 sec
5	13:25:59	SELECT * FROM question_service_db.question LIMIT 0, 1000	0 row(s) returned	0.078 sec / 0.000 sec
6	15:08:33	SELECT * FROM quizservice.question LIMIT 0, 1000	0 row(s) returned	0.032 sec / 0.000 sec
7	15:09:18	SELECT * FROM quizserver_db.question LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
8	15:09:29	SELECT * FROM quizserver_db.quiz LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

- one_2_one_mapping
- question_service_db
 - question
 - quiz
 - quiz_questions
 - Views
 - Stored Procedures
 - Functions
- quizserver
 - quizserver_db
 - Tables
 - question
 - quiz
 - quiz_questions
 - Views
 - Stored Procedures
 - Functions
 - sakila
 - student
 - sys
 - world

Administration Schemas

Information

Schema: question_service_db

SQL File 10* SQL File 11* question quiz SQL File 4* question question quiz quiz_questions

1 • SELECT * FROM quizserver_db.quiz_questions;

Result Grid

quiz_id	questions_question_id
1	3

Output

#	Time	Action	Message	Duration / Fetch
4	13.25.45	SELECT * FROM one_2_one_mapping table LIMIT 0, 1000	3 row(s) returned	0.031 sec / 0.000 sec
5	13.25.59	SELECT * FROM question_service_db question LIMIT 0, 1000	0 row(s) returned	0.078 sec / 0.000 sec
6	15.08.33	SELECT * FROM quizservice question LIMIT 0, 1000	0 row(s) returned	0.032 sec / 0.000 sec
7	15.09.18	SELECT * FROM quizserver_db question LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
8	15.09.29	SELECT * FROM quizserver_db quiz LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
9	15.09.37	SELECT * FROM quizserver_db quiz_questions LIMIT 0, 1000	1 row(s) returned	0.015 sec / 0.000 sec

Object Info Session

localhost:8080/swagger-ui/swagger-ui/index.html

Swagger

/api-docs

Explore

OpenAPI definition v0 OAS 3.0

/api-docs

Servers

http://localhost:8080 - Generated server url

quiz-controller

- POST /api/v1/quiz/submitQuiz/{id}
- POST /api/v1/quiz/create
- GET /api/v1/quiz/getQuizById/{id}

question-controller

- POST /api/v1/question/addNewQuestion

