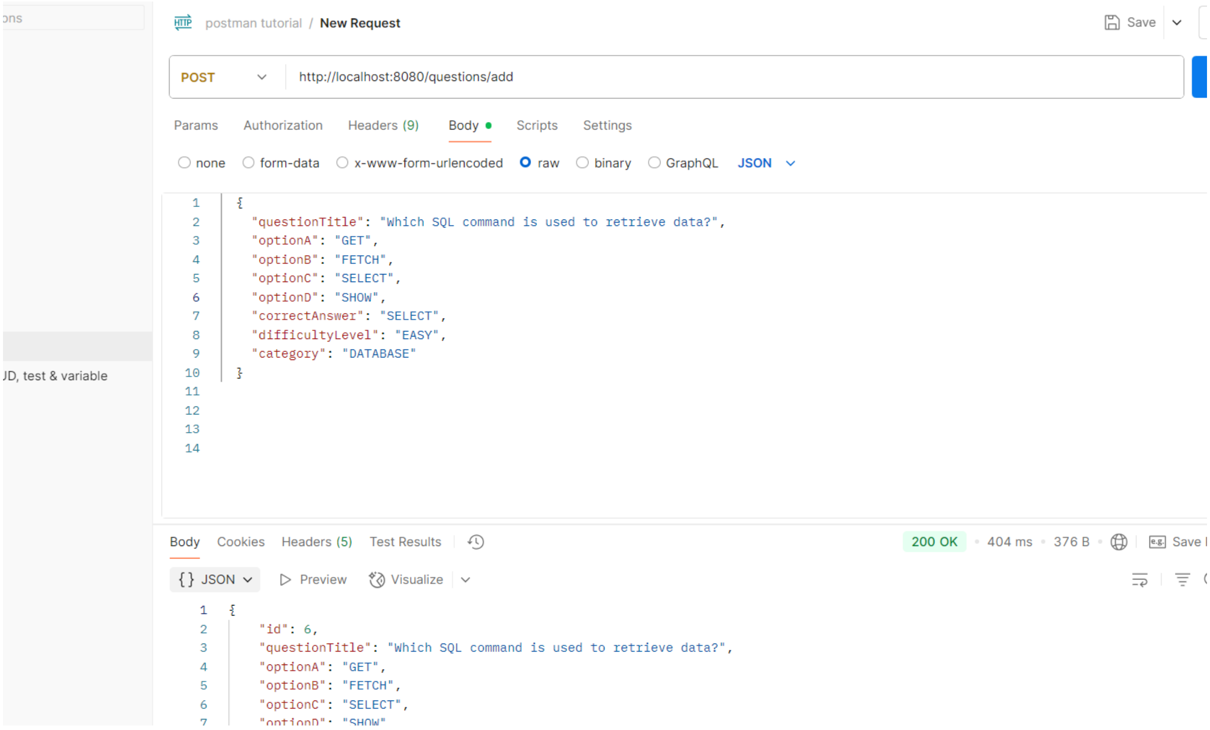


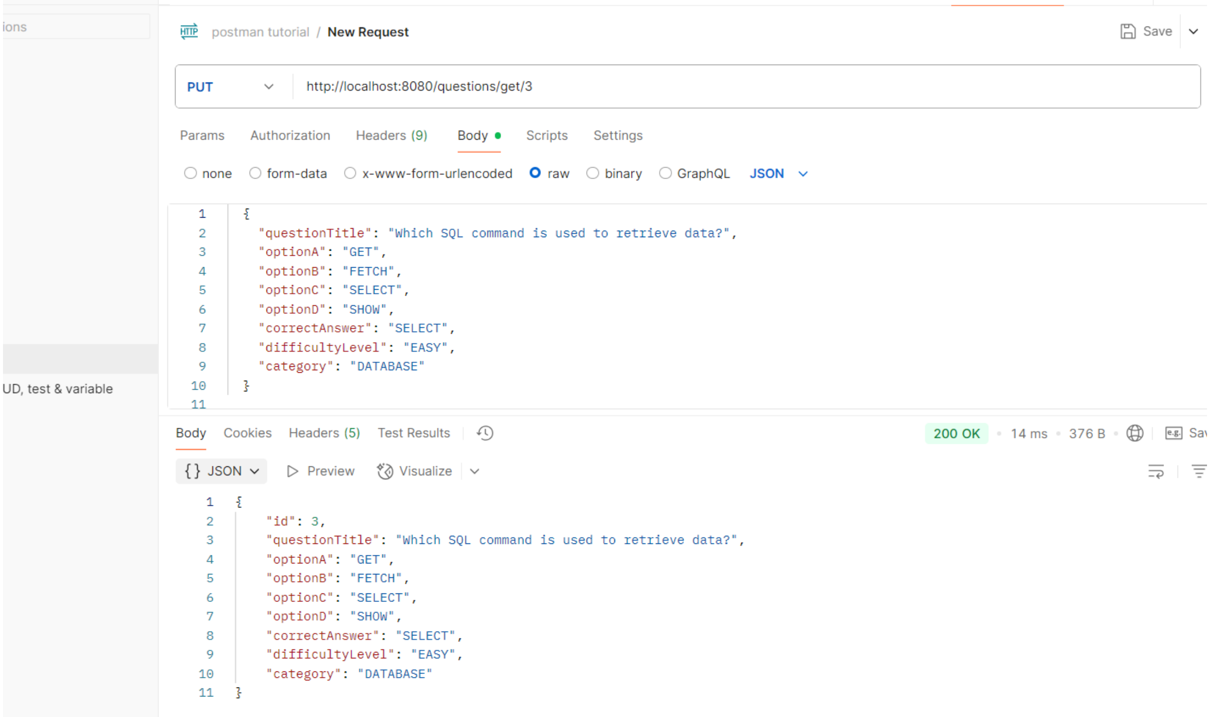
POST http://localhost:8080/questions/add

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK • 404 ms • 376 B

{} JSON

```
{
    "id": 6,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW"
```



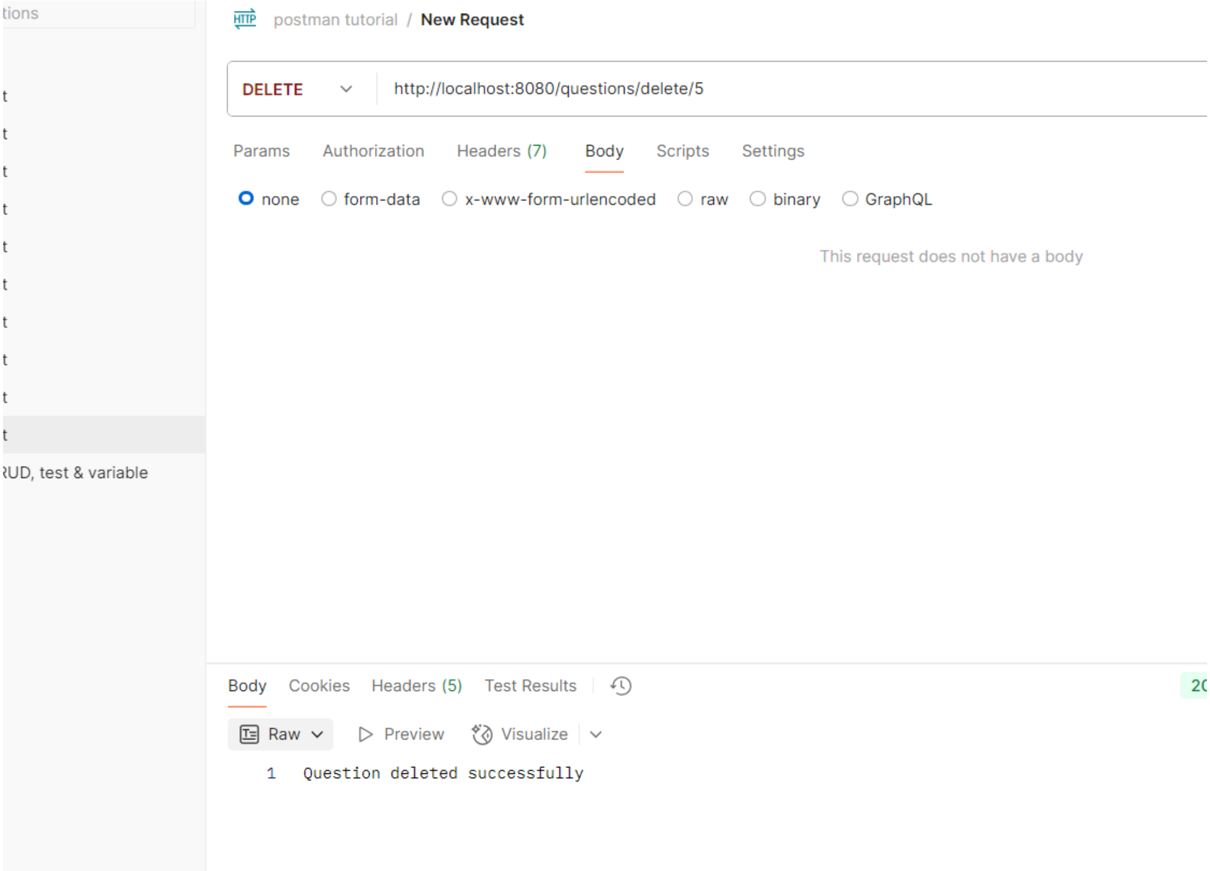
PUT http://localhost:8080/questions/get/3

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK • 14 ms • 376 B

{} JSON

```
{
    "id": 3,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW",
    "correctAnswer": "SELECT",
    "difficultyLevel": "EASY",
    "category": "DATABASE"
}
```

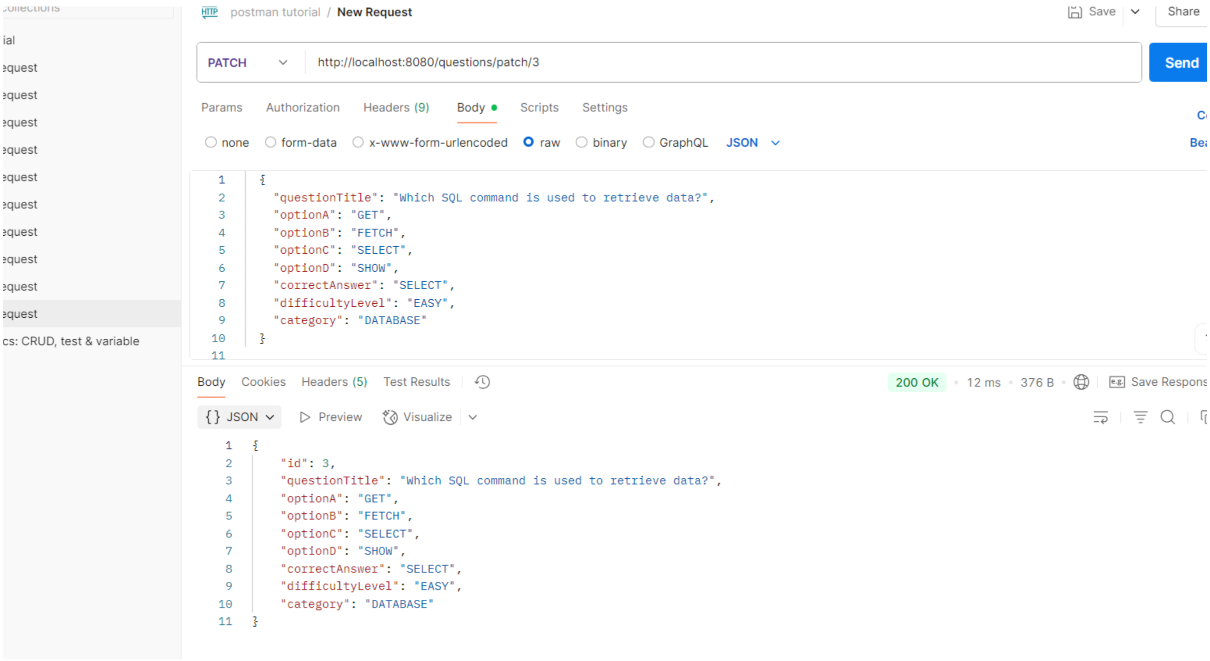


DELETE http://localhost:8080/questions/delete/5

Body: ● none

This request does not have a body

```
Question deleted successfully
```



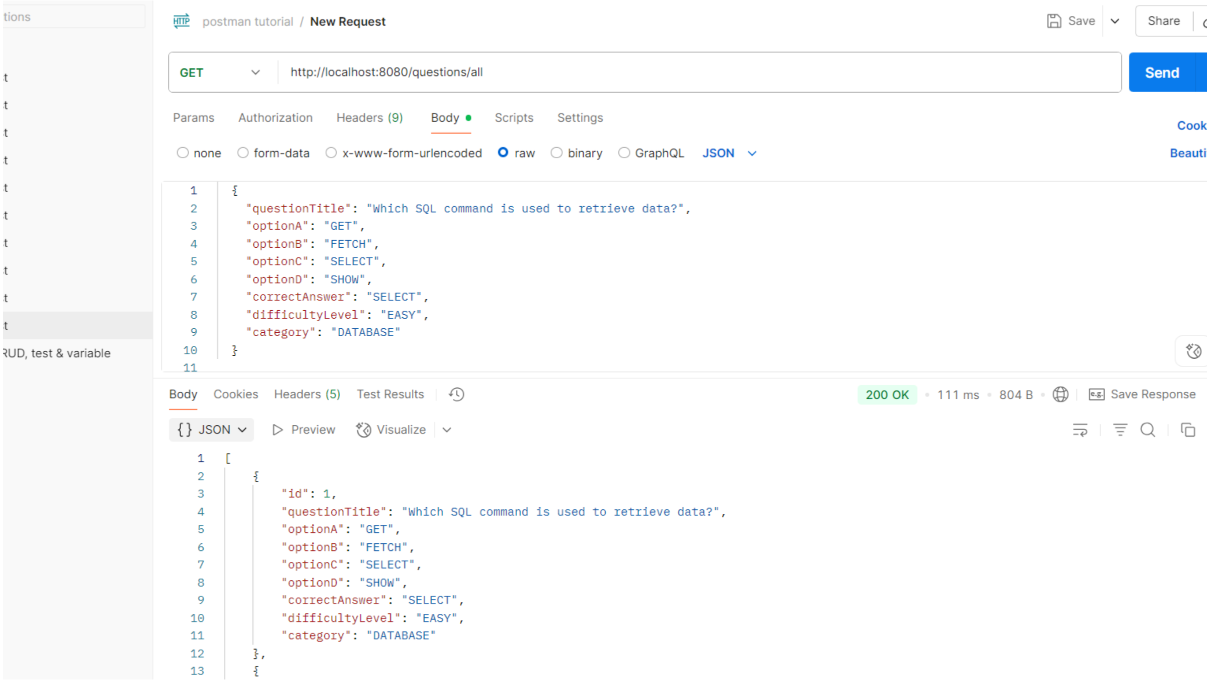
PATCH http://localhost:8080/questions/patch/3

Body: ● raw JSON

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK · 12 ms · 376 B

```
{
    "id": 3,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW",
    "correctAnswer": "SELECT",
    "difficultyLevel": "EASY",
    "category": "DATABASE"
}
```

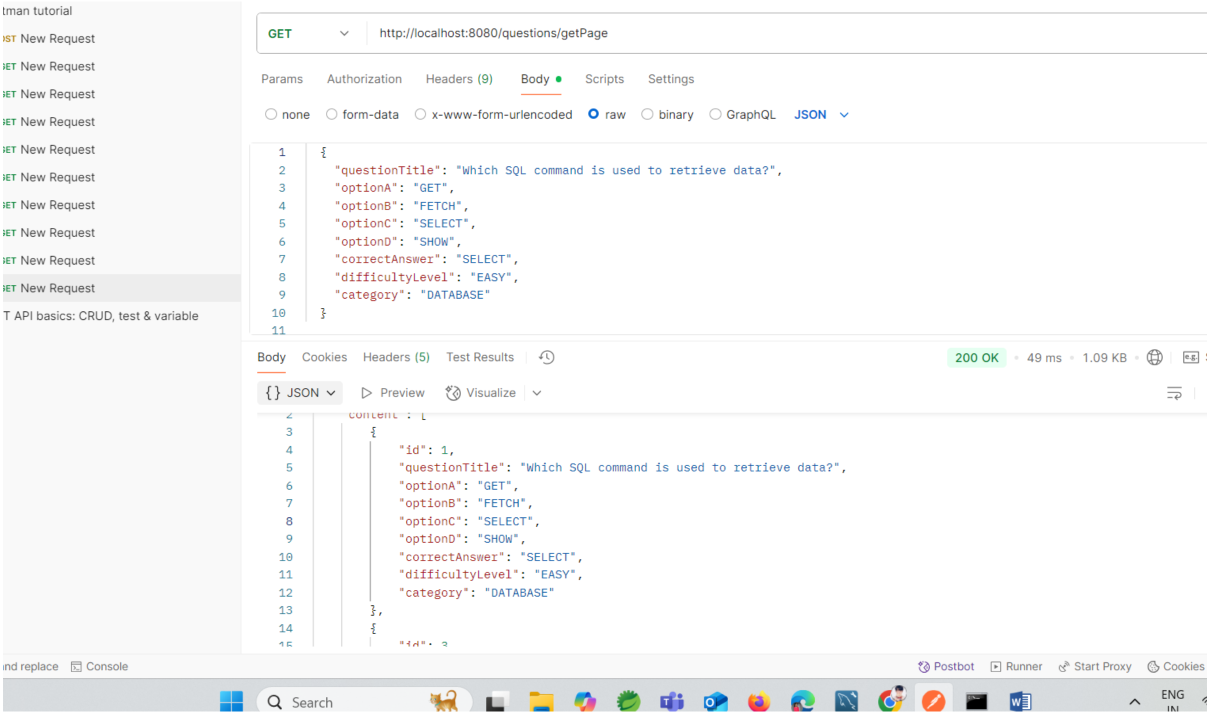


GET http://localhost:8080/questions/all

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK • 111 ms • 804 B

```
[
    {
        "id": 1,
        "questionTitle": "Which SQL command is used to retrieve data?",
        "optionA": "GET",
        "optionB": "FETCH",
        "optionC": "SELECT",
        "optionD": "SHOW",
        "correctAnswer": "SELECT",
        "difficultyLevel": "EASY",
        "category": "DATABASE"
    },
    {
```

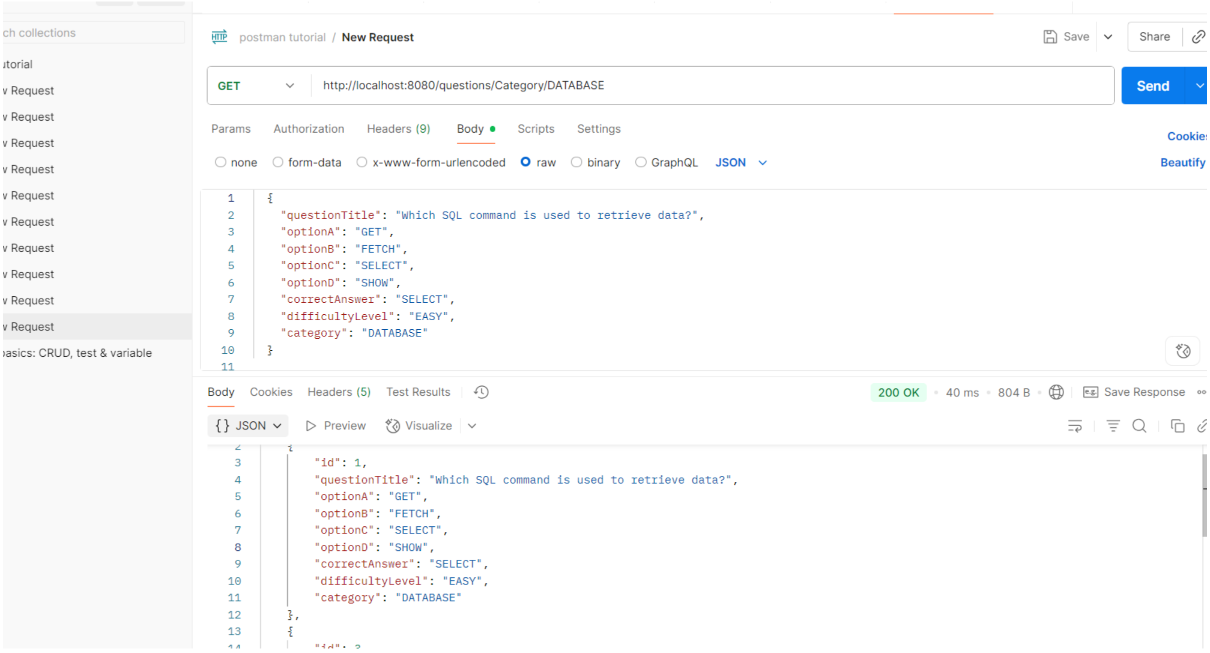


GET http://localhost:8080/questions/getPage

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK • 49 ms • 1.09 KB

```
    "content": [
        {
            "id": 1,
            "questionTitle": "Which SQL command is used to retrieve data?",
            "optionA": "GET",
            "optionB": "FETCH",
            "optionC": "SELECT",
            "optionD": "SHOW",
            "correctAnswer": "SELECT",
            "difficultyLevel": "EASY",
            "category": "DATABASE"
        },
        {
            "id": 3
```



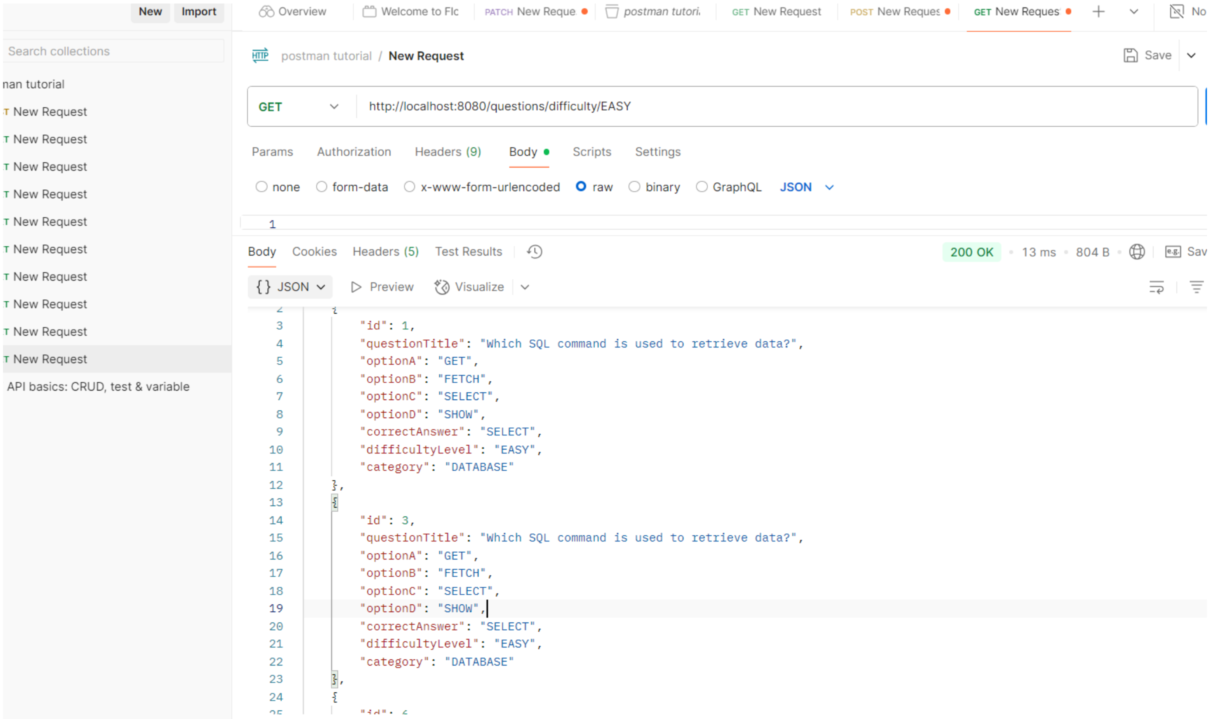
GET http://localhost:8080/questions/Category/DATABASE

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
  "questionTitle": "Which SQL command is used to retrieve data?",
  "optionA": "GET",
  "optionB": "FETCH",
  "optionC": "SELECT",
  "optionD": "SHOW",
  "correctAnswer": "SELECT",
  "difficultyLevel": "EASY",
  "category": "DATABASE"
}
```

200 OK · 40 ms · 804 B

```
    "id": 1,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW",
    "correctAnswer": "SELECT",
    "difficultyLevel": "EASY",
    "category": "DATABASE"
  },
  {
```

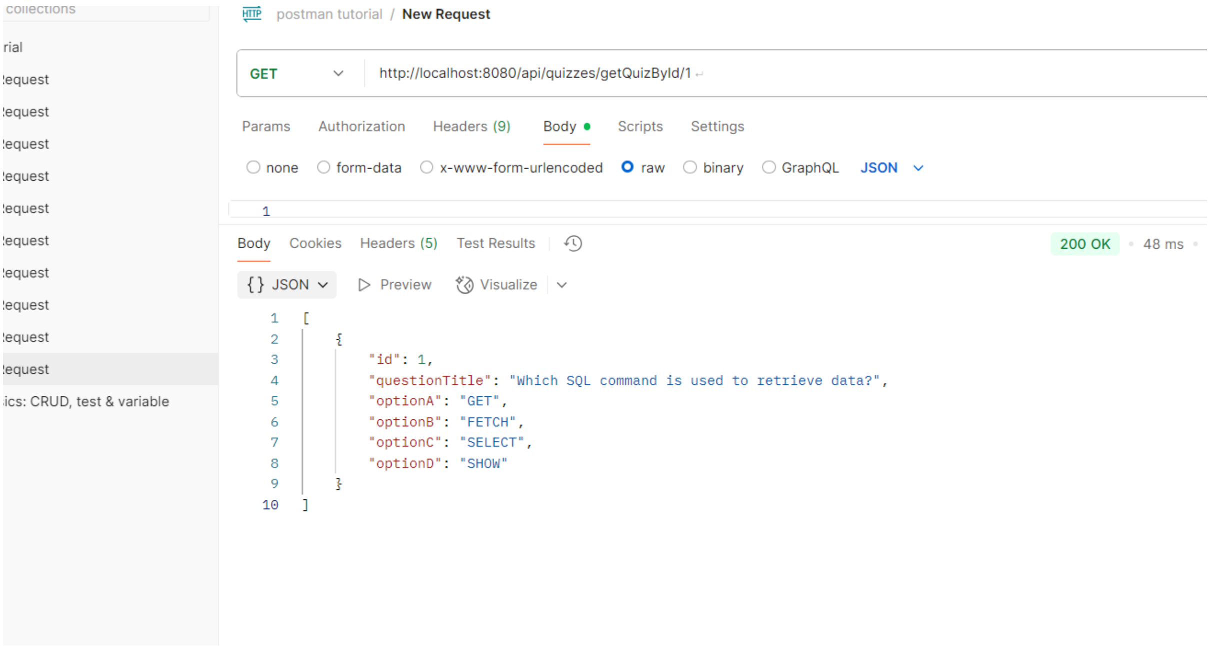


GET http://localhost:8080/questions/difficulty/EASY

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

200 OK · 13 ms · 804 B

```
    "id": 1,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW",
    "correctAnswer": "SELECT",
    "difficultyLevel": "EASY",
    "category": "DATABASE"
  },
  {
    "id": 3,
    "questionTitle": "Which SQL command is used to retrieve data?",
    "optionA": "GET",
    "optionB": "FETCH",
    "optionC": "SELECT",
    "optionD": "SHOW",
    "correctAnswer": "SELECT",
    "difficultyLevel": "EASY",
    "category": "DATABASE"
  },
  {
```

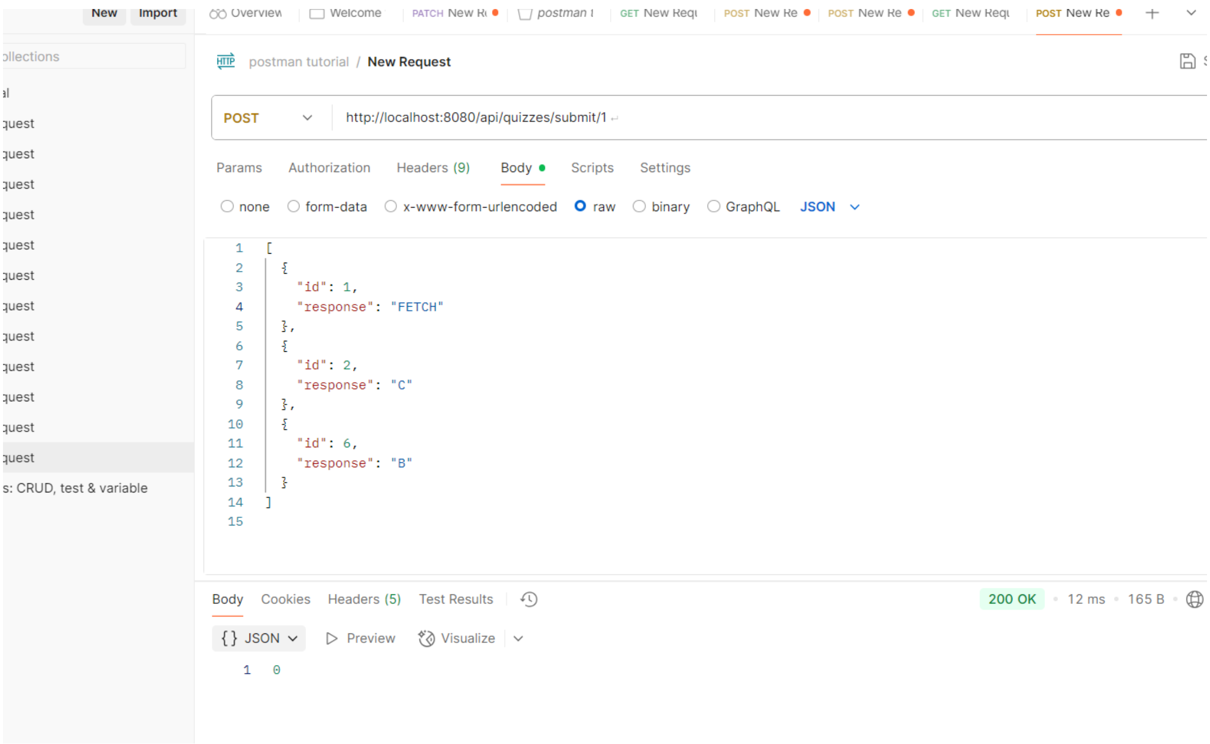


postman tutorial / New Request

GET http://localhost:8080/api/quizzes/getQuizById/1

200 OK • 48 ms

```
[
    {
        "id": 1,
        "questionTitle": "Which SQL command is used to retrieve data?",
        "optionA": "GET",
        "optionB": "FETCH",
        "optionC": "SELECT",
        "optionD": "SHOW"
    }
]
```



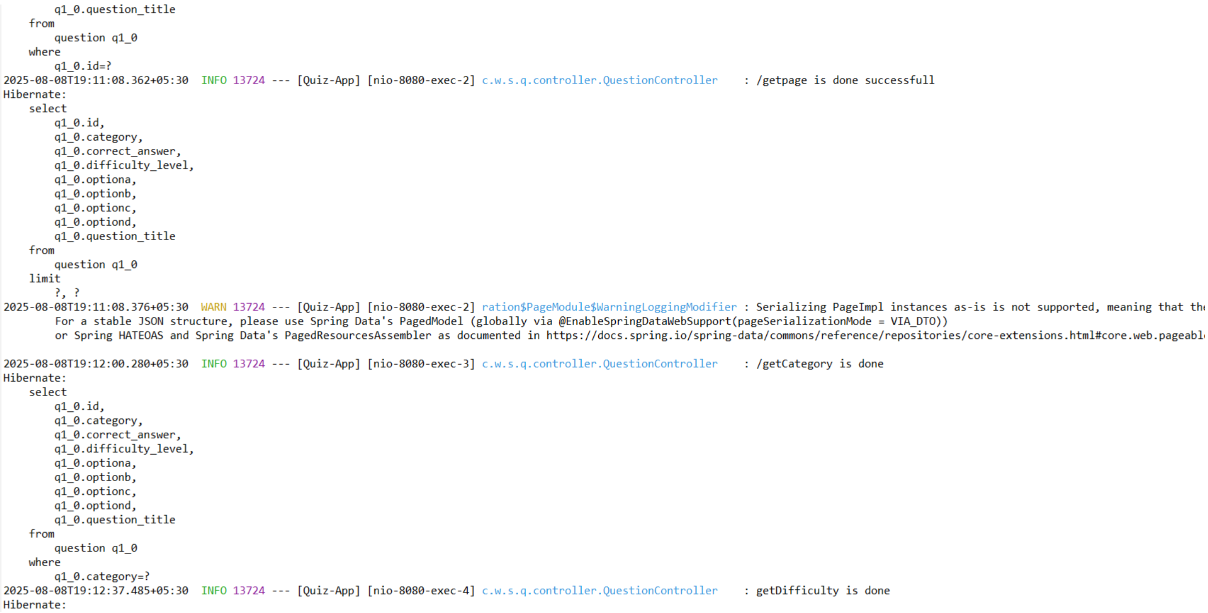
postman tutorial / New Request

POST http://localhost:8080/api/quizzes/submit/1

```
[
  {
    "id": 1,
    "response": "FETCH"
  },
  {
    "id": 2,
    "response": "C"
  },
  {
    "id": 6,
    "response": "B"
  }
]
```

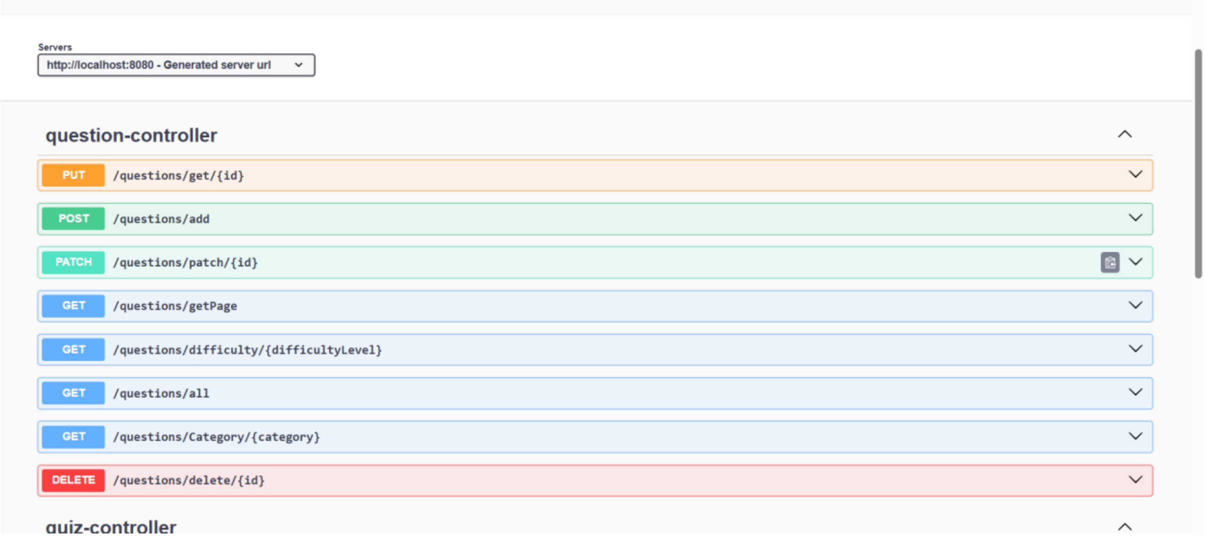
200 OK • 12 ms • 165 B

```
0
```

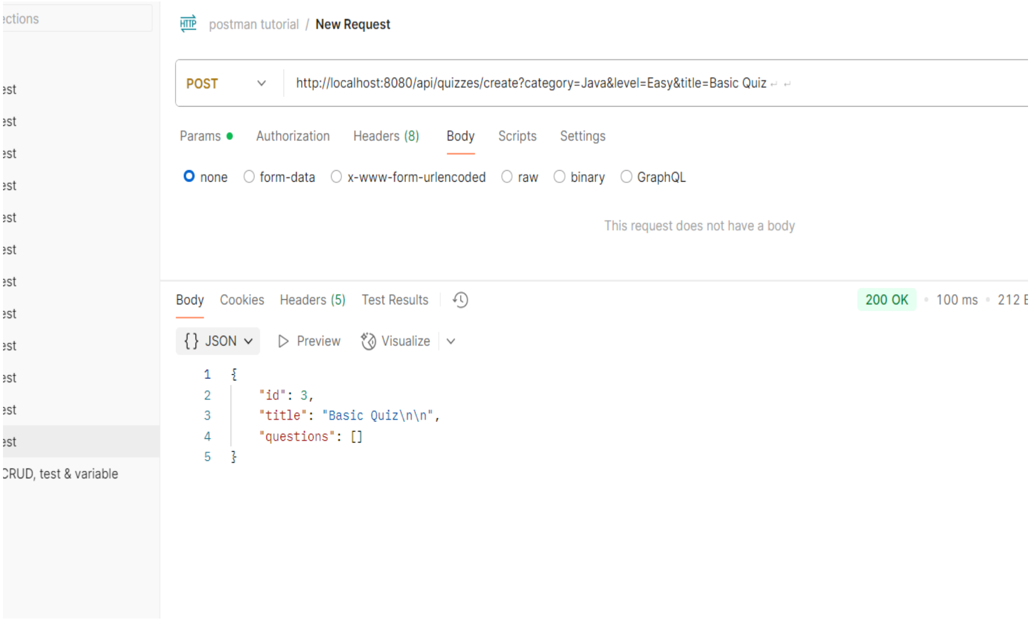


```
        q1_0.question_title
    from
        question q1_0
    where
        q1_0.id=?
2025-08-08T19:11:08.362+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-2] c.w.s.q.controller.QuestionController    : /getpage is done successfull
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
        q1_0.question_title
    from
        question q1_0
    limit
        ?, ?
2025-08-08T19:11:08.376+05:30  WARN 13724 --- [Quiz-App] [nio-8080-exec-2] ration$PageModule$WarningLoggingModifier : Serializing PageImpl instances as-is is not supported, meaning that th
        For a stable JSON structure, please use Spring Data's PagedModel (globally via @EnableSpringDataWebSupport(pageSerializationMode = VIA_DTO))
        or Spring HATEOAS and Spring Data's PagedResourcesAssembler as documented in https://docs.spring.io/spring-data/commons/reference/repositories/core-extensions.html#core.web.pageabl

2025-08-08T19:12:00.280+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-3] c.w.s.q.controller.QuestionController    : /getCategory is done
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
        q1_0.question_title
    from
        question q1_0
    where
        q1_0.category=?
2025-08-08T19:12:37.485+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-4] c.w.s.q.controller.QuestionController    : getDifficulty is done
Hibernate: 
```



Servers
http://localhost:8080 - Generated server url
question-controller
PUT /questions/get/{id}
POST /questions/add
PATCH /questions/patch/{id}
GET /questions/getPage
GET /questions/difficulty/{difficultyLevel}
GET /questions/all
GET /questions/Category/{category}
DELETE /questions/delete/{id}
quiz-controller



ctions
est
est
est
est
est
est
est
est
est
est
est
est
CRUD, test & variable
postman tutorial / New Request
Params ● Authorization Headers (8) Body Scripts Settings
Body Cookies Headers (5) Test Results
{} JSON ▷ Preview Visualize

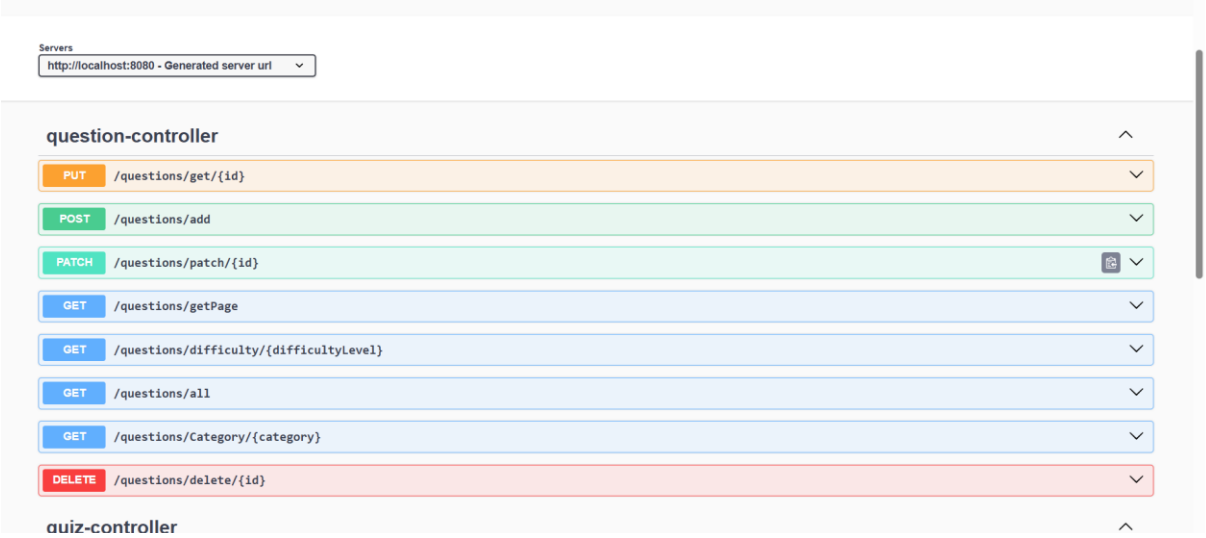
POST http://localhost:8080/api/quizzes/create?category=Java&level=Easy&title=Basic Quiz

● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

200 OK · 100 ms · 212 B

```
{
    "id": 3,
    "title": "Basic Quiz\n\n",
    "questions": []
}
```



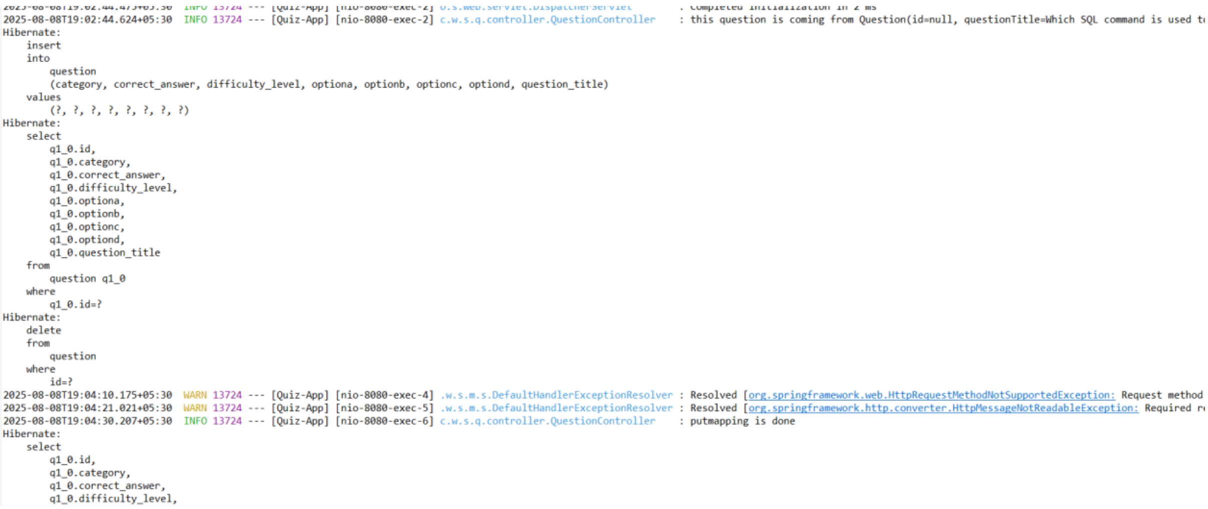
Servers

http://localhost:8080 - Generated server url

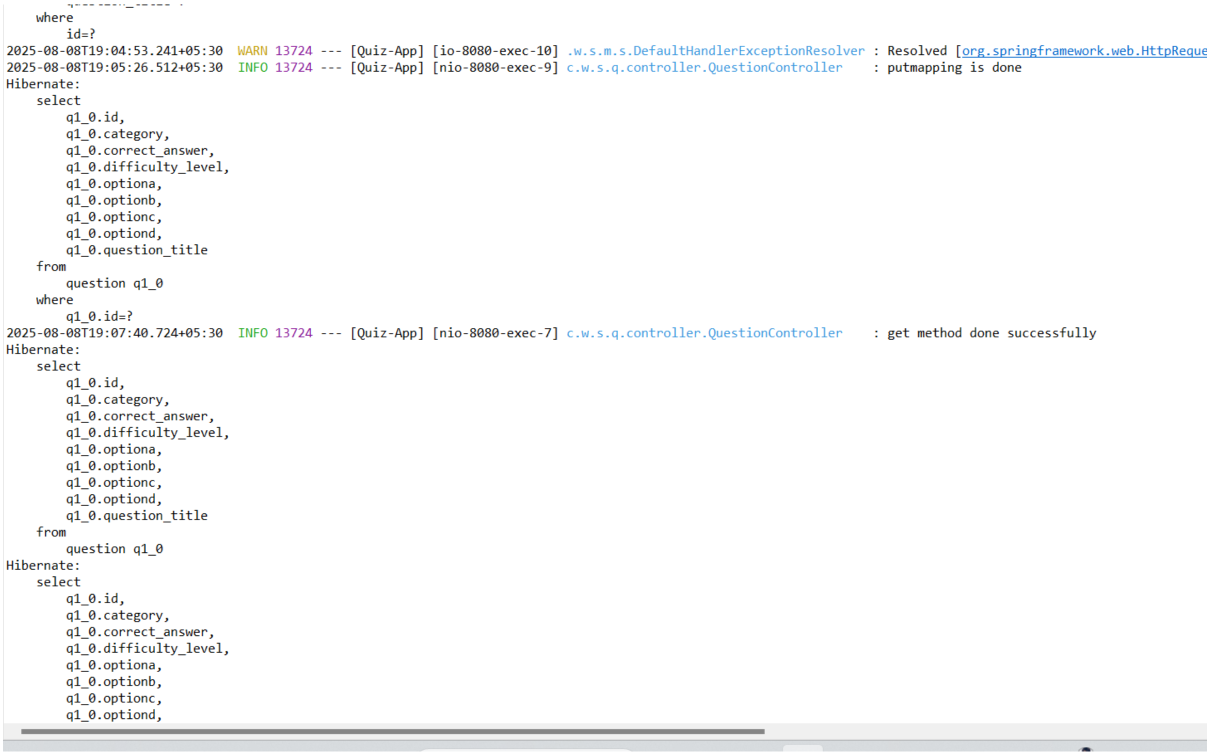
question-controller

PUT	/questions/get/{id}
POST	/questions/add
PATCH	/questions/patch/{id}
GET	/questions/getPage
GET	/questions/difficulty/{difficultyLevel}
GET	/questions/all
GET	/questions/Category/{category}
DELETE	/questions/delete/{id}

quiz-controller



```
2025-08-08T19:02:44.475+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-2] o.s.web.servlet.DispatcherServlet        : Completed initialization in 2 ms
2025-08-08T19:02:44.624+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-2] c.w.s.q.controller.QuestionController    : this question is coming from Question(id=null, questionTitle=Which SQL command is used t
Hibernate: 
    insert 
    into
        question
        (category, correct_answer, difficulty_level, optiona, optionb, optionc, optiond, question_title) 
    values
        (?, ?, ?, ?, ?, ?, ?, ?)
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
        q1_0.question_title 
    from
        question q1_0 
    where
        q1_0.id=?
Hibernate: 
    delete 
    from
        question 
    where
        id=?
2025-08-08T19:04:10.175+05:30  WARN 13724 --- [Quiz-App] [nio-8080-exec-4] .w.s.m.s.DefaultHandlerExceptionResolver : Resolved [org.springframework.web.HttpRequestMethodNotSupportedException: Request method
2025-08-08T19:04:21.021+05:30  WARN 13724 --- [Quiz-App] [nio-8080-exec-5] .w.s.m.s.DefaultHandlerExceptionResolver : Resolved [org.springframework.http.converter.HttpMessageNotReadableException: Required r
2025-08-08T19:04:30.207+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-6] c.w.s.q.controller.QuestionController    : putmapping is done
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
```



```
    where
        id=?
2025-08-08T19:04:53.241+05:30  WARN 13724 --- [Quiz-App] [io-8080-exec-10] .w.s.m.s.DefaultHandlerExceptionResolver : Resolved [org.springframework.web.HttpReque
2025-08-08T19:05:26.512+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-9] c.w.s.q.controller.QuestionController    : putmapping is done
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
        q1_0.question_title 
    from
        question q1_0 
    where
        q1_0.id=?
2025-08-08T19:07:40.724+05:30  INFO 13724 --- [Quiz-App] [nio-8080-exec-7] c.w.s.q.controller.QuestionController    : get method done successfully
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
        q1_0.question_title 
    from
        question q1_0
Hibernate: 
    select
        q1_0.id,
        q1_0.category,
        q1_0.correct_answer,
        q1_0.difficulty_level,
        q1_0.optiona,
        q1_0.optionb,
        q1_0.optionc,
        q1_0.optiond,
```