

1. Add Question (POST)

GET http://localhost:8080/api/

POST http://localhost:8080/ap

POST http://localhost:8080/ap

+

...

HTTP

http://localhost:8080/api/v1/question/add

POST

http://localhost:8080/api/v1/question/add

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

JSON

```
3  "optionA": "5Java",
4  "optionB": "Java5",
5  "optionC": "23Java",
6  "optionD": "Error",
7  "correctAnswer": "A",
8  "category": "JAVA",
9  "difficultyLevel": "EASY"
10 }
11
```

Body

Cookies

Headers (5)

Test Results

Status: 200 OK

Pretty

Raw

Preview

Visualize

JSON

```
1  {
2    "id": 1,
3    "title": "What is the output of System.out.println(2 + 3 + \"Java\")?",
4    "optionA": "5Java",
5    "optionB": "Java5",
6    "optionC": "23Java",
7    "optionD": "Error",
8    "correctAnswer": "A",
9    "category": "JAVA",
10   "difficultyLevel": "EASY"
11 }
```

2. Get Paginated Questions (GET)

Postman interface showing a GET request to `http://localhost:8080/api/v1/question?page=0&size=10`. The response is a JSON object with a `pageable` object and a list of questions.

```
1 {
2   "title": "What is the output of System.out.println(2 + 3 + \"Java\")?",
3   "optionA": "5Java",
4   "optionB": "Java5",
5   "optionC": "23Java",
6   "optionD": "Error",
7   "correctAnswer": "A",
8   "category": "JAVA",
9 }
10
11 {
12   "pageable": {
13     "pageNumber": 0,
14     "pageSize": 20,
15     "sort": {
16       "sorted": false,
17       "unsorted": true,
18       "empty": true
19     },
20     "offset": 0,
21     "unpaged": false,
22     "paged": true
23   },
24   "questions": [
25     {
26       "id": 1,
27       "title": "What is the output of System.out.println(2 + 3 + \"Java\")?",
28       "optionA": "5Java",
29       "optionB": "Java5",
30       "optionC": "23Java",
31       "optionD": "Error",
32       "correctAnswer": "A",
33       "category": "JAVA",
34       "difficultyLevel": "EASY"
35     },
36     {
37       "id": 2,
38       "title": "What is the size of int in Java?",
39       "optionA": "2 bytes",
40       "optionB": "4 bytes",
41       "optionC": "8 bytes",
42       "optionD": "Depends on OS",
43       "correctAnswer": "B",
44       "category": "JAVA",
45       "difficultyLevel": "EASY"
46     }
47   ]
48 }
```

3. Get Questions by Category (GET)

Postman interface showing a GET request to `http://localhost:8080/api/v1/question/category/JAVA`. The response is a JSON array of two question objects.

```
1 [
2   {
3     "id": 1,
4     "title": "What is the output of System.out.println(2 + 3 + \"Java\")?",
5     "optionA": "5Java",
6     "optionB": "Java5",
7     "optionC": "23Java",
8     "optionD": "Error",
9     "correctAnswer": "A",
10    "category": "JAVA",
11    "difficultyLevel": "EASY"
12  },
13  {
14    "id": 2,
15    "title": "What is the size of int in Java?",
16    "optionA": "2 bytes",
17    "optionB": "4 bytes",
18    "optionC": "8 bytes",
19    "optionD": "Depends on OS",
20    "correctAnswer": "B",
21    "category": "JAVA",
22    "difficultyLevel": "EASY"
23  }
24 ]
```

4. Create Quiz (POST)

The screenshot shows a Postman interface with a POST request to `http://localhost:8080/api/v1/quiz/create?category=JAVA&difficulty=EASY`. The request body is a JSON object: `{ "quizId": 1, "message": "Quiz created successfully" }`. The response status is 200 OK, with a time of 543 ms and a size of 214 B. The response body is also a JSON object: `{ "quizId": 1, "message": "Quiz created successfully" }`.

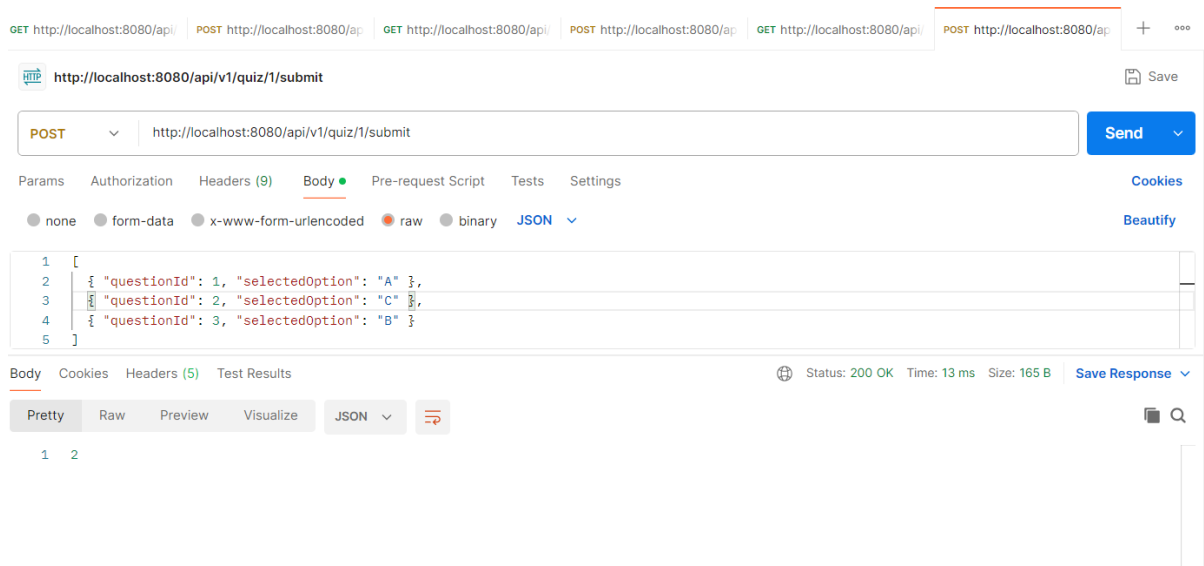
```
1 {
2   "quizId": 1,
3   "message": "Quiz created successfully"
4 }
```

Get quiz questions (GET):

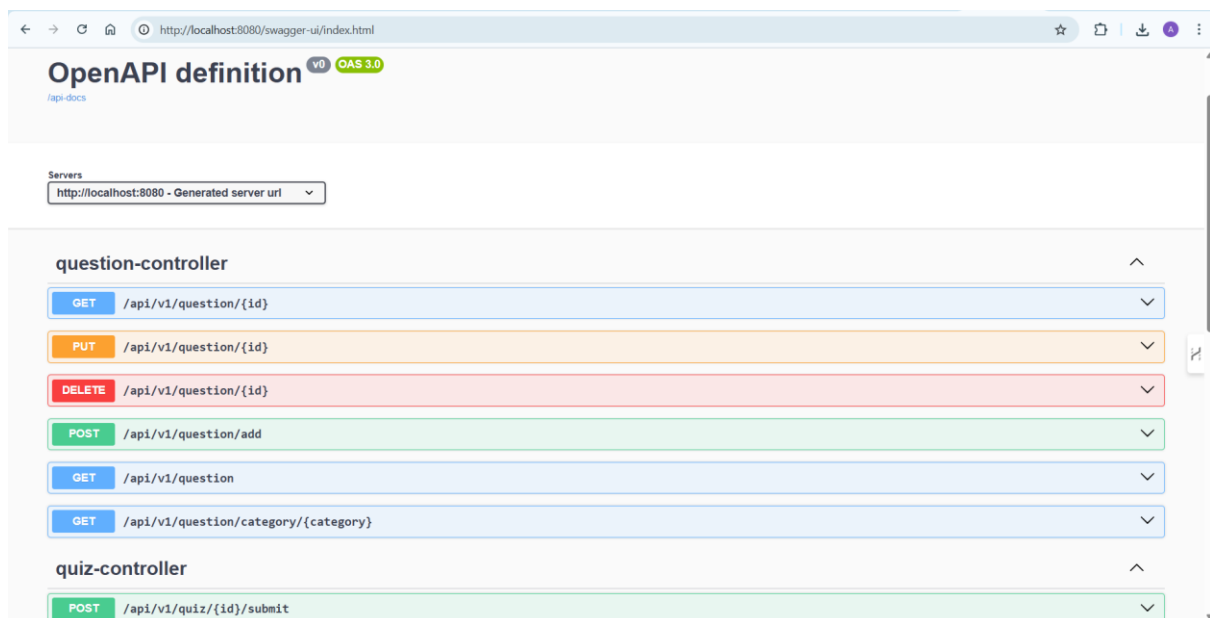
The screenshot shows a Postman interface with a GET request to `http://localhost:8080/api/v1/quiz/1/questions`. The response status is 200 OK, with a time of 99 ms and a size of 592 B. The response body is a JSON array of two quiz questions.

```
1 [
2   {
3     "id": 2,
4     "title": "Which keyword is used to inherit a class in Java?",
5     "optionA": "super",
6     "optionB": "this",
7     "optionC": "extends",
8     "optionD": "implements"
9   },
10  {
11    "id": 1,
12    "title": "What is the size of int in Java?",
13    "optionA": "2 bytes",
14    "optionB": "4 bytes",
15    "optionC": "8 bytes",
16    "optionD": "Depends on OS"
17  }
18 ]
```

Submit quiz (POST)



/swagger-ui – An interactive web interface that lets you **view and test your API endpoints** directly in the browser.



[illegible]