

# Quiz Monolithic Repo POSTMAN API Requests

## 1 Create Quiz Questions

The screenshot displays the Postman interface with a new HTTP request configured. The URL is `http://localhost:8080/saveQuestions` and the method is `POST`. The request body is set to `raw` and contains the following JSON:

```
1 {
2   "questionTitle": "What is 200 ok means?",
3   "option1": "Created",
4   "option2": "Success",
5   "option3": "Bad Request",
6   "option4": "Can use for anything , but standard is Success",
7   "correctAnswer": "Can use for anything , but standard is Success",
8   "difficultyLevel": "EASY",
9   "category": "JAVA"
10 }
11
```

The response is shown in the bottom panel, indicating a `200 OK` status with a response time of `216 ms` and a size of `443 B`. The response body is displayed in `JSON` format:

```
1 [{"id":1,"questionTitle":"What is 200 ok means?","option1":"Created","option2":"Success","option3":"Bad Request","option4":"Can use for anything , but
  standard is Success","correctAnswer":"Can use for anything , but standard is Success","difficultyLevel":"EASY","category":"JAVA"}]
```

The interface includes a sidebar with navigation icons, a top bar with search and window controls, and a bottom status bar with links to Launchpad, Live Share, and Finish Setup.

## 2 GEt All questions

The screenshot shows the Postman application interface. The top bar indicates the current request is a GET to `http://localhost:8080/getAllQuestions`. The left sidebar shows the 'My Workspace' and a list of requests, including 'REST API basics: ...'. The main panel is divided into tabs: 'Params', 'Authorization', 'Headers (7)', 'Body', 'Scripts', and 'Settings'. The 'Body' tab is active, showing the response in 'Pretty' format. The response is a JSON array of two objects, each representing a question with its title, options, correct answer, difficulty level, and category.

Query Params

Key	Value
Key	Value

Body

Status: 200 OK Time: 144 ms Size: 676 B

```
1 [{"id":1,"questionTitle":"What is 200 ok means?","option1":"Created","option2":"Success","option3":"Bad Request","option4":"Can use for anything , but standard is Success","correctAnswer":"Can use for anything , but standard is Success","difficultyLevel":"EASY","category":"JAVA"}, {"id":2,"questionTitle":"What is the command to push git code?","option1":"git init","option2":"git add ","option3":"git commit -m message","option4":"git push","correctAnswer":"git push","difficultyLevel":"EASY","category":"GIT"}]
```

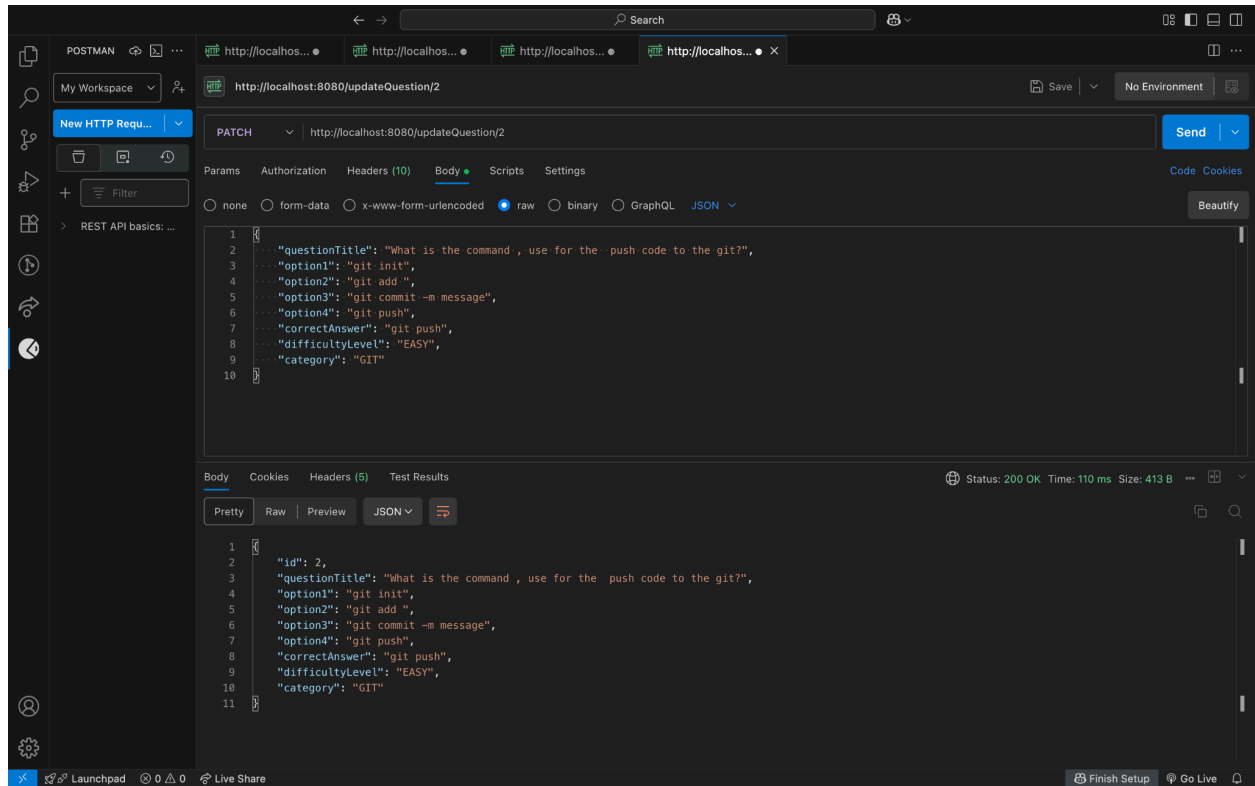
### 3 Get questions by id

The screenshot shows the Postman interface with a GET request to `http://localhost:8080/getQuestionsById/2` successfully executed. The response is a JSON object with the following structure:

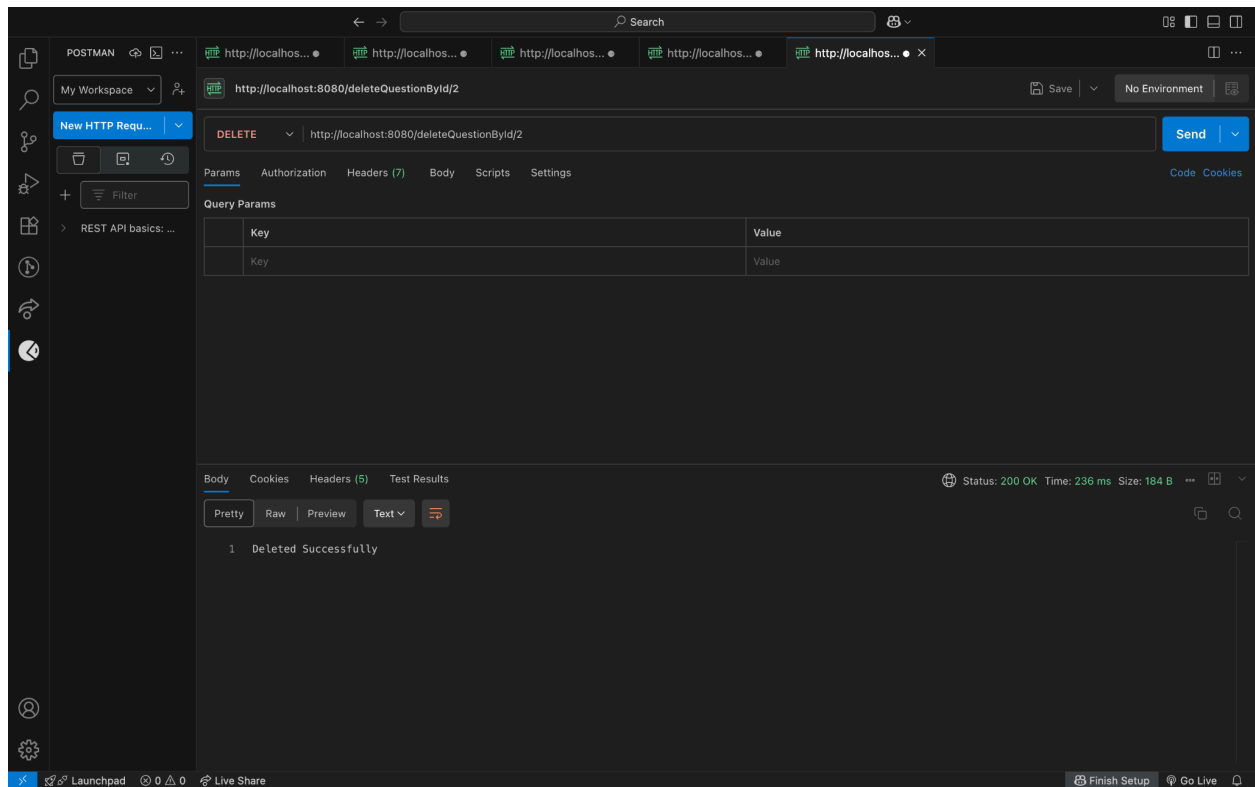
```
1 {
2   "id": 2,
3   "questionTitle": "What is the command to push git code?",
4   "option1": "git init",
5   "option2": "git add ",
6   "option3": "git commit -m message",
7   "option4": "git push",
8   "correctAnswer": "git push",
9   "difficultyLevel": "EASY",
10  "category": "GIT"
11 }
```

The status bar at the bottom indicates a 200 OK response with a time of 35 ms and a size of 394 B. The bottom of the interface shows a Launchpad icon, a status of 0/0, a Live Share button, and a Finish Setup button.

## 4 Update question by id



## 5 Delete question by id



## 6 Create Quiz and get quiz

The screenshot displays the Postman REST client interface. The top section shows a POST request to the endpoint `http://localhost:8080/quiz/create?category=JAVA&level=EASY&title=JavaQuiz1`. The request body is a JSON object with the following structure:

```
1 {
2   "id": 1,
3   "title": "JavaQuiz1",
4   "questions": [
5     {
6       "id": 1, "questionTitle": "What is JVM?", "option1": "Java Virtual Machine", "option2": "Java Variable Method", "option3": "Just Virtual Memory",
7       "option4": "Joint Visual Model", "correctAnswer": "Java Virtual Machine", "difficultyLevel": "EASY", "category": "JAVA"
8     }
9   ]
10 }
```

The bottom section shows the response body, which is a JSON array of quiz objects. The first object in the array is:

```
1 {
2   "id": 10, "title": "JavaQuiz1", "questions": [
3     {
4       "id": 7, "questionTitle": "Which keyword is used to inherit a class in Java?", "option1": "implement",
5       "option2": "extends", "option3": "inherits", "option4": "super", "correctAnswer": "extends", "difficultyLevel": "EASY", "category": "JAVA"
6     },
7     {
8       "id": 4, "questionTitle": "Which keyword is used to inherit a class in Java?", "option1": "implement", "option2": "extends", "option3": "inherits", "option4": "super",
9       "correctAnswer": "extends", "difficultyLevel": "EASY", "category": "JAVA"
10    },
11    {
12      "id": 8, "questionTitle": "Which keyword is used to inherit a class in Java?", "option1": "implement", "option2": "extends", "option3": "inherits", "option4": "super",
13      "correctAnswer": "extends", "difficultyLevel": "EASY", "category": "JAVA"
14    },
15    {
16      "id": 1, "questionTitle": "What is 200 ok means?", "option1": "Created", "option2": "Success", "option3": "Bad Request", "option4": "Can use for anything , but standard is Success",
17      "correctAnswer": "Can use for anything , but standard is Success", "difficultyLevel": "EASY", "category": "JAVA"
18    },
19    {
20      "id": 6, "questionTitle": "Which keyword is used to inherit a class in Java?", "option1": "implement", "option2": "extends", "option3": "inherits", "option4": "super",
21      "correctAnswer": "extends", "difficultyLevel": "EASY", "category": "JAVA"
22    }
23  ]
24 }
```

The status bar at the bottom indicates a successful response with status 200 OK, a time of 309 ms, and a size of 1.36 KB.

POSTMAN interface showing a REST client request to `http://localhost:8080/quiz/get/10/questions`. The request is a GET method. The response status is 200 OK, Time: 53 ms, Size: 969 B.

Query Params:

Key	Value
Key	Value

Body (JSON):

```
1 [{"id":7,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":4,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":8,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":1,"questionTitle":"What is 200 ok means?","option1":"Created","option2":"Success","option3":"Bad Request","option4":"Can use for anything , but standard is Success"}, {"id":6,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}]
```

POSTMAN interface showing a REST client request to `http://localhost:8080/quiz/get/10/questions`. The request is a GET method. The response status is 200 OK, Time: 53 ms, Size: 969 B.

Query Params:

Key	Value
Key	Value

Body (JSON):

```
1 [{"id":7,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":4,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":8,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}, {"id":1,"questionTitle":"What is 200 ok means?","option1":"Created","option2":"Success","option3":"Bad Request","option4":"Can use for anything , but standard is Success"}, {"id":6,"questionTitle":"Which keyword is used to inherit a class in Java?","option1":"implement","option2":"extends","option3":"inherits","option4":"super"}]
```