

## Method GET

The screenshot shows the Postman interface with a GET request to `http://localhost:8000/books`. The response is a JSON object with a status of 200 OK, a message "Success to get Book", and a list of three books. The left sidebar shows the "My Workspace" with a collection named "belajar\_API" containing a folder "section19" which is expanded to show the selected request.

```
1 {
2   "code": 200,
3   "message": "Success to get Book",
4   "status": "Success",
5   "data": [
6     {
7       "id": 1,
8       "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
9       "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto",
10      "user_id": 0
11     },
12     {
13       "id": 2,
14       "title": "qui est esse",
15       "body": "et iusto tempore vitae\nequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla",
16       "user_id": 0
17     },
18     {
19       "id": 3,
20       "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
21       "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut",
22       "user_id": 0
23     },
24   ]
25 }
```

## Method GET 3

The screenshot shows the Postman interface with a GET request to `http://localhost:8000/books/3`. The response is a JSON object with a status of 200 OK, a message "Success to get Book", and a single book object. The left sidebar shows the "My Workspace" with a collection named "belajar\_API" containing a folder "section19" which is expanded to show the selected request.

```
1 {
2   "code": 200,
3   "message": "Success to get Book",
4   "status": "Success",
5   "data": {
6     "id": 3,
7     "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
8     "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut",
9     "user_id": 0
10   }
11 }
```

## Method POST

Postman interface showing a POST request to `http://localhost:8000/books`. The request is configured with the following body:

Key	Value	Description
title	glat membaca	
body	apa itu membaca	
user_id	1	

The response is displayed in the Body tab, showing a successful status (200 OK) and the following JSON data:

```
1 {
2   "code": 200,
3   "message": "Success to get Book",
4   "status": "Success",
5   "data": {
6     "id": 101,
7     "title": "glat membaca",
8     "body": "apa itu membaca",
9     "user_id": 1
10  }
11 }
```

## Method DELETE

Postman interface showing a DELETE request to `http://localhost:8000/books/101`. The request is configured with the following body:

Key	Value	Description
Key	Value	Description

The response is displayed in the Body tab, showing a successful status (200 OK) and the following JSON data:

```
1 {
2   "code": 200,
3   "message": "Success to delete Book",
4   "status": "Success",
5   "data": {
6     "id": 0,
7     "title": "",
8     "body": "",
9     "user_id": 0
10  }
11 }
```