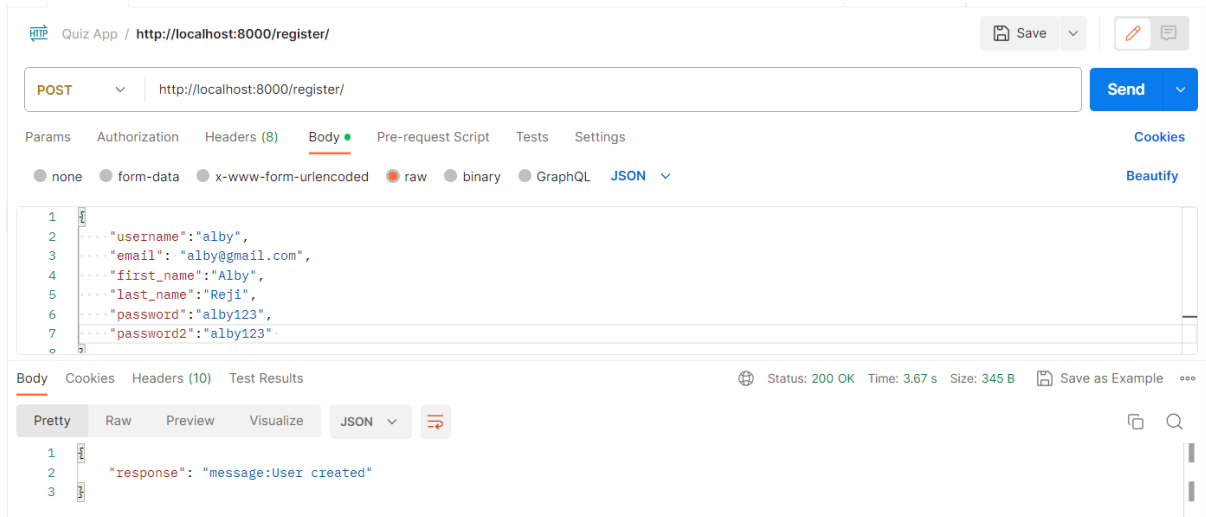


# Blogging platform

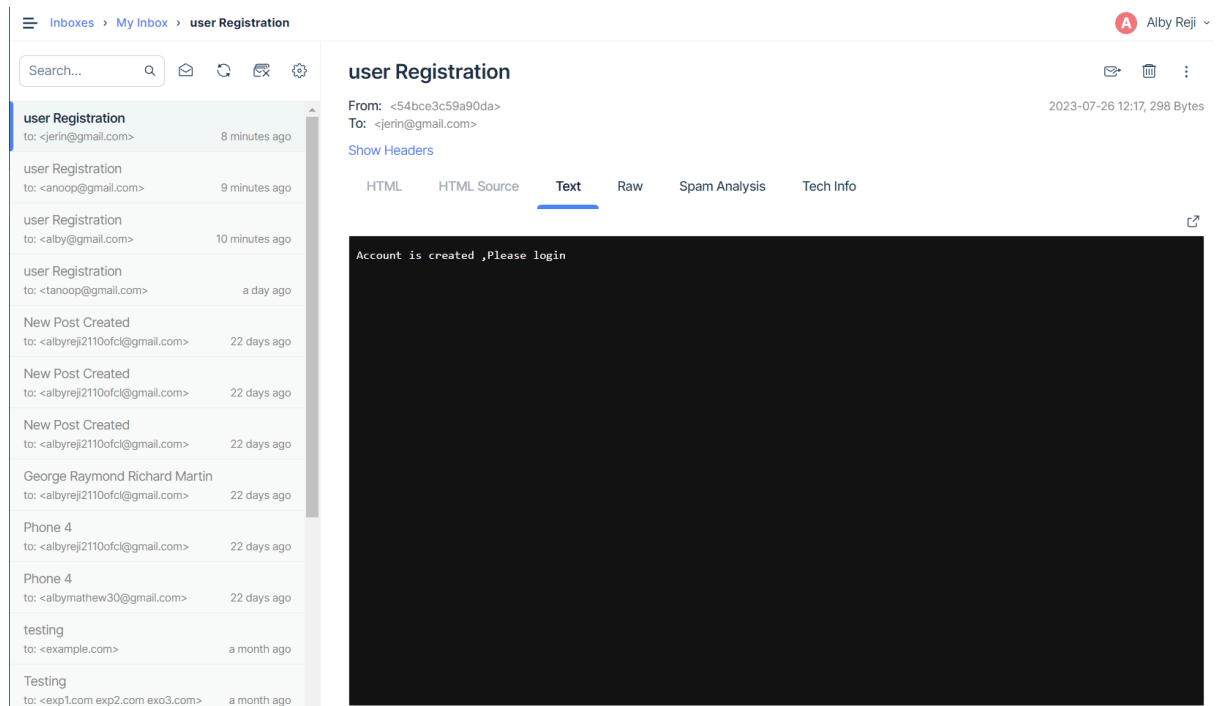
## User section

### 1. User Authentication

- Registration



- Mail Confirmation



• Login

Quiz App / http://localhost:8000/login/

Save

POST

http://localhost:8000/login/

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	email	jerin@gmail.com			
<input checked="" type="checkbox"/>	password	jerin123			
	Key	Value	Description		

Body

Cookies

Headers (10)

Test Results

Status: 200 OK

Time: 418 ms

Size: 794 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

Content-Type: application/json

"refresh": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2t1b190eX8lIjoicmVmcmVzaCI6ImV4cCI6MTY5MDQ2MDQ0MywiaWF0IjoxNjkwMzc0MDQzLCJqdGkiOiJlZjBjNTEyZjU4MDc0NDI0YWUyOTQ1OTUwMjk2OWQwZCIsInVzZXZfaWQ1OjR9.8UV48d8p8OT\_1dFXuG1WQet3i\_2wKVbTHR\_Y6zqrDrk",

"access": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2t1b190eX8lIjoicmVmcmVzaCI6ImV4cCI6MTY5MDQ2MDQ0MywiaWF0IjoxNjkwMzc1MjQzLCJqdGkiOiJlZjBjNTEyZjU4MDc0NDI0YWUyOTQ1OTUwMjk2OWQwZCIsInVzZXZfaWQ1OjR9.8UV48d8p8OT\_1dFXuG1WQet3i\_2wKVbTHR\_Y6zqrDrk"

## 2. Blog Posts

- Create blogs.

The screenshot shows a REST client interface with the URL `http://localhost:8000/bloglistcreate/`. The request method is `POST`. The request body is set to `form-data` and contains three fields: `blog_title` with value `Virat Kohli`, `blog_content` with value `Virat Kohli is an Indian international cricketer.`, and `image` with value `2.png`. The response status is `201 Created` with a time of `1339 ms` and size of `595 B`. The response body is a JSON object:

```
{
  "id": 1,
  "author": "jerin",
  "blog_title": "Virat Kohli",
  "blog_content": "Virat Kohli is an Indian international cricketer.",
  "created_time": "2023-07-26T12:30:32.856923Z",
  "updated_time": "2023-07-26T12:30:32.856923Z",
  "image": "http://localhost:8000/media/blog_images/2.png",
  "comments": []
}
```

- Retrieve Blog

The screenshot shows a REST client interface with the URL `http://localhost:8000/blogdetail/1/`. The request method is `GET`. The request is authenticated using a Bearer Token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1b290IjoiMj02IiwiaWF0IjoiMTY5MzUyMzUyIn0.`. The response status is `200 OK` with a time of `20 ms` and size of `615 B`. The response body is a JSON object:

```
{
  "id": 1,
  "author": "jerin",
  "blog_title": "Virat Kohli",
  "blog_content": "Virat Kohli is an Indian international cricketer.",
  "created_time": "2023-07-26T12:30:32.856923Z",
  "updated_time": "2023-07-26T12:30:32.856923Z",
  "image": "http://localhost:8000/media/blog_images/2.png",
  "comments": []
}
```

## • Put

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8000/blogdetail/1/`
- Method:** `PUT`
- Authorization:** Bearer Token. The token is `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`
- Body:** JSON data:

```
{  "id": 1,  "author": "jerin",  "blog_title": "Virat Kohli - Update",  "blog_content": "Virat Kohli is an Indian international cricketer.",  "created_time": "2023-07-26T12:30:32.856923Z",  "updated_time": "2023-07-26T12:49:05.818171Z",  "image": "http://localhost:8000/media/blog_images/2_JxajNhs.png",  "comments": []}
```
- Status:** 200 OK, Time: 396 ms, Size: 632 B


## • Patch


The screenshot shows a REST client interface with the following details:



- URL:** `http://localhost:8000/blogdetail/1/`
- Method:** `PATCH`
- Authorization:** Bearer Token. The token is `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`
- Body:** JSON data:


```
{  "id": 1,  "author": "jerin",  "blog_title": "Virat Kohli - Update 1",  "blog_content": "Virat Kohli is an Indian international cricketer.",  "created_time": "2023-07-26T12:30:32.856923Z",  "updated_time": "2023-07-26T12:48:46.374716Z",  "image": "http://localhost:8000/media/blog_images/2.png",  "comments": []}
```
- Status:** 200 OK, Time: 142 ms, Size: 626 B


- Delete


 http://localhost:8000/blogdetail/4/ 

Save 


DELETE  http://localhost:8000/blogdetail/4/ 

Send 



Params **Authorization**  Headers (7) Body Pre-request Script Tests Settings 

Cookies

Type 

Bearer Token 


The authorization header will be automatically generated when you send the request. Learn more about [authorization](#) ↗


 Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about [variables](#) ↗ 

Token 


eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...


Body Cookies Headers (10) Test Results 

 Status: 200 OK Time: 111 ms Size: 375 B 

Save as Example 

Pretty Raw Preview Visualize 


JSON 






1 2 3


```
"message": "Blog deleted successfully."
```


- List Of Blogs


 http://localhost:8000/bloglist/ 

Save 


GET  http://localhost:8000/bloglist/ 

Send 



Params **Authorization**  Headers (7) Body Pre-request Script Tests Settings 

Cookies

Type 

Bearer Token 


The authorization header will be automatically generated when you send the request. Learn more about [authorization](#) ↗


 Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about [variables](#) ↗ 

Token 


eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...


Body Cookies Headers (10) Test Results 

 Status: 200 OK Time: 209 ms Size: 1.17 KB 

Save as Example 

Pretty Raw Preview Visualize 

JSON 



6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

```
{
  "id": 1,
  "author": "jerin",
  "blog_title": "Virat Kohli",
  "blog_content": "Virat Kohli is an Indian international cricketer.",
  "created_time": "2023-07-26T12:30:32.856923Z",
  "updated_time": "2023-07-26T12:30:32.856923Z",
  "image": "http://localhost:8000/media/blog_images/2.png",
  "comments": []
},
{
  "id": 2,
  "author": "alby",
  "blog_title": "Rohit Sharma",
  "blog_content": "Rohit Sharma is an Indian international cricketer.",
  "created_time": "2023-07-26T12:35:02.755105Z",
  "updated_time": "2023-07-26T12:35:02.755105Z",
  "image": "http://localhost:8000/media/blog_images/6.png",
  "comments": []
}
```

### 3. Comments

- Add Comment

The screenshot shows a REST client interface with the URL `http://localhost:8000/commentcreate/1/`. The request method is `POST`. The request body is in `form-data` format with the following fields:

Key	Value	Description
<input checked="" type="checkbox"/> <code>blog_comment</code>	King Kohli	
<input type="checkbox"/> <code>blog_post</code>	1	
Key	Value	Description

The response status is `201 Created` with a time of `127 ms` and a size of `456 B`. The response body is in `JSON` format and contains the following data:


```
1 {
2   "id": 2,
3   "user": "alby",
4   "blog_comment": "King Kohli",
5   "created_time": "2023-07-26T14:29:00.620149Z",
6   "updated_time": "2023-07-26T14:29:00.620149Z"
7 }
```

- List of Comments

The screenshot shows a REST client interface with the URL `http://localhost:8000/commentlist/1/`. The request method is `GET`. The request is authenticated using a `Bearer Token` with the token `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`. The response status is `200 OK` with a time of `161 ms` and a size of `651 B`. The response body is in `JSON` format and contains the following data:

```
1 {
2   "count": 2,
3   "next": null,
4   "previous": null,
5   "results": [
6     {
7       "id": 1,
8       "user": "jeirin",
9       "blog_comment": "Good Player",
10      "created_time": "2023-07-26T14:27:43.019746Z",
11      "updated_time": "2023-07-26T14:27:43.019746Z"
12     },
13     {
14       "id": 2,
15       "user": "alby",
16       "blog_comment": "King Kohli",
17       "created_time": "2023-07-26T14:29:00.620149Z",
18       "updated_time": "2023-07-26T14:29:00.620149Z"
19     }
20   ]
21 }
```

- Edit Comments

 http://localhost:8000/commentdetail/2/

PUT

http://localhost:8000/commentdetail/2/

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

2

3

... "blog\_comment": "Good Player...Amazing Player"

Body

Cookies

Headers (10)

Test Results

Status: 200 OK

Time: 282 ms

Size: 494 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

7

"id": 2,


"user": "alby",

"blog\_comment": "Good Player...Amazing Player",

"created\_time": "2023-07-26T14:29:00.620149Z",

"updated\_time": "2023-07-26T14:35:25.114702Z"

- Delete Comments

 http://localhost:8000/commentdetail/3/

DELETE

http://localhost:8000/commentdetail/3/

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Type

Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...

Body

Cookies

Headers (10)

Test Results

Status: 200 OK

Time: 153 ms

Size: 378 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

1

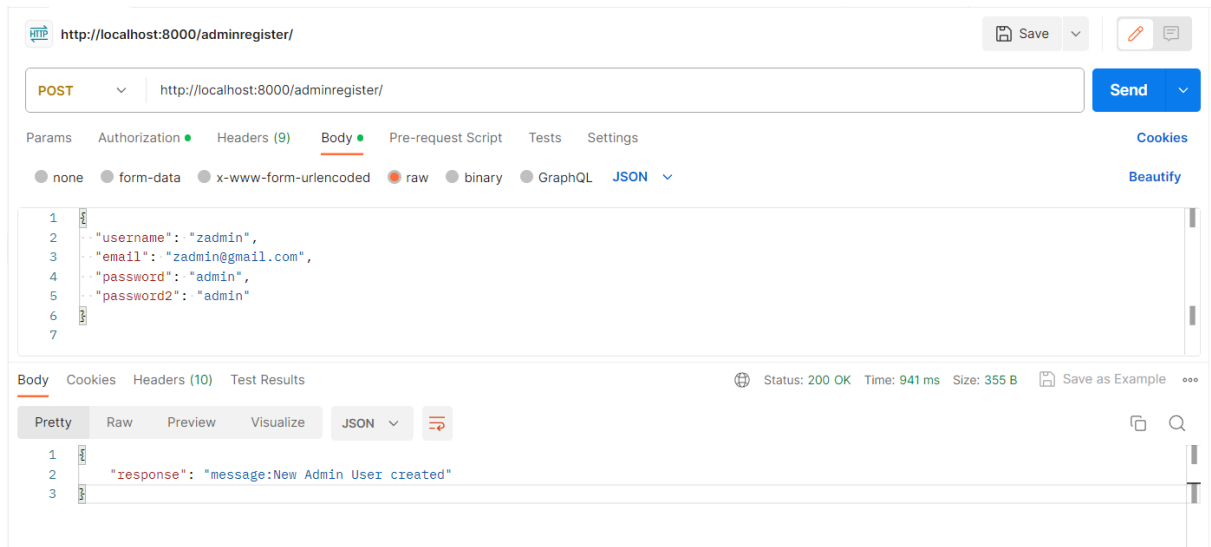
2

3

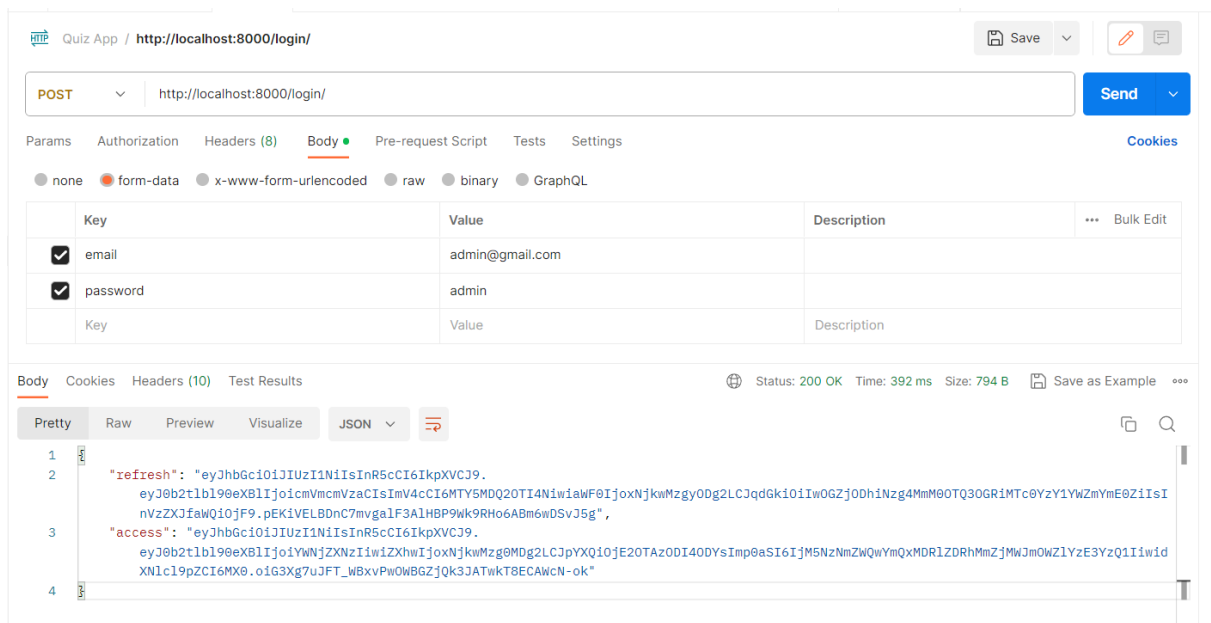
"message": "Comment deleted successfully."

## Admin Section

- New Admin Registration



- Admin Login





# 1. Blog Posts

- List of blogs

REST client interface showing a GET request to `http://localhost:8000/adminlist/` with a Bearer Token authorization header. The response is a JSON array of 3 blog posts.

Request:

- Method: GET
- URL: `http://localhost:8000/adminlist/`
- Authorization: Bearer Token
- Token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`

Response (JSON):

```
1 {
2   "count": 3,
3   "next": null,
4   "previous": null,
5   "results": [
6     {
7       "id": 1,
8       "author": "jerin",
9       "blog_title": "Virat Kohli - Update",
10      "blog_content": "Virat Kohli is an Indian international cricketer.",
11      "created_time": "2023-07-26T12:30:32.856923Z",
12      "updated_time": "2023-07-26T12:49:05.818171Z",
13      "image": "http://localhost:8000/media/blog_images/2_JxaJNhs.png"
14    },
15    {
16      "id": 2,
17      "author": "alby",
18      "blog_title": "Rohit Sharma",
19      "blog_content": "Rohit Sharma is an Indian international cricketer.",
20      "created_time": "2023-07-26T12:35:02.755105Z",
21      "updated_time": "2023-07-26T12:35:02.755105Z",
22      "image": "http://localhost:8000/media/blog_images/6.png"
23    }
24  ]
25 }
```

- Delete Blogs

REST client interface showing a DELETE request to `http://localhost:8000/admindetail/3/` with a Bearer Token authorization header. The response is a JSON object with a success message.

Request:

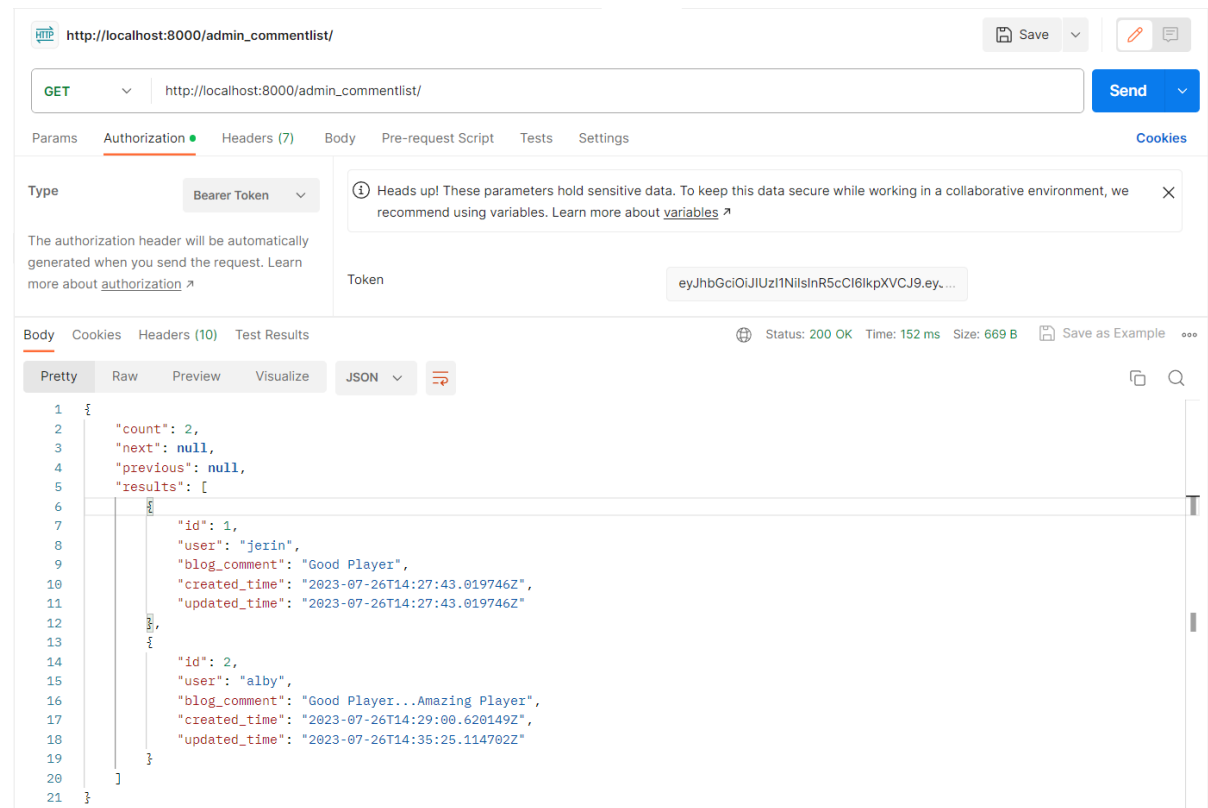
- Method: DELETE
- URL: `http://localhost:8000/admindetail/3/`
- Authorization: Bearer Token
- Token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`

Response (JSON):

```
1 {
2   "message": "Blog deleted successfully."
3 }
```

## 2. Comments

- List of blogs



REST client interface showing a GET request to `http://localhost:8000/admin_commentlist/`. The request is authenticated with a Bearer Token. The response is a JSON array of two comment objects.

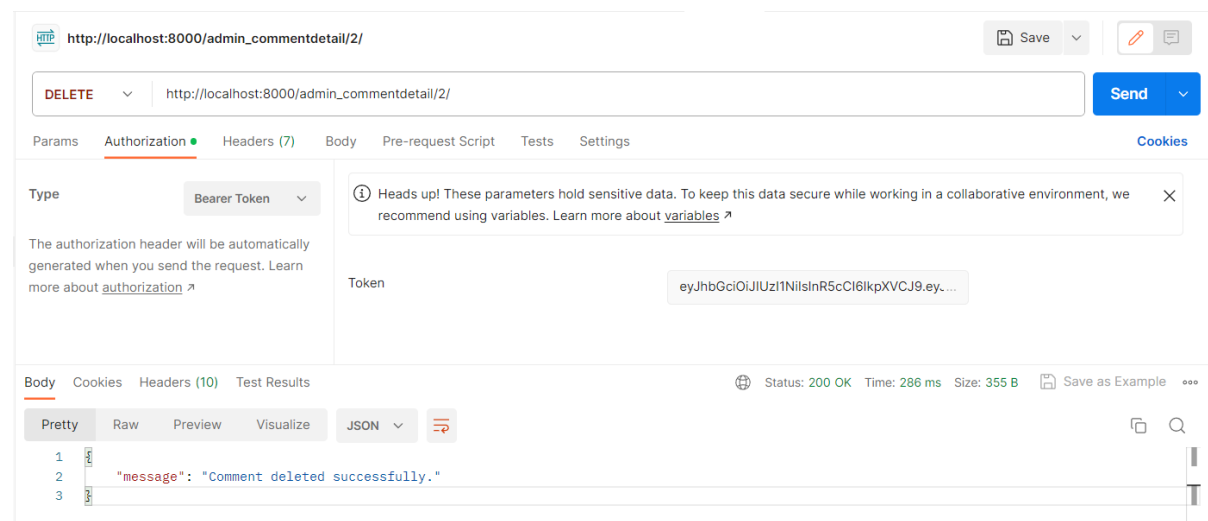
**Request:**

- Method: GET
- URL: `http://localhost:8000/admin_commentlist/`
- Authorization: Bearer Token
- Token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`

**Response (JSON):**

```
1 {
2   "count": 2,
3   "next": null,
4   "previous": null,
5   "results": [
6     {
7       "id": 1,
8       "user": "jerin",
9       "blog_comment": "Good Player",
10      "created_time": "2023-07-26T14:27:43.019746Z",
11      "updated_time": "2023-07-26T14:27:43.019746Z"
12     },
13     {
14       "id": 2,
15       "user": "alby",
16       "blog_comment": "Good Player...Amazing Player",
17       "created_time": "2023-07-26T14:29:00.620149Z",
18       "updated_time": "2023-07-26T14:35:25.114702Z"
19     }
20   ]
21 }
```

- Delete Comments



REST client interface showing a DELETE request to `http://localhost:8000/admin_commentdetail/2/`. The request is authenticated with a Bearer Token. The response is a JSON object indicating successful deletion.

**Request:**

- Method: DELETE
- URL: `http://localhost:8000/admin_commentdetail/2/`
- Authorization: Bearer Token
- Token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`

**Response (JSON):**

```
1 {
2   "message": "Comment deleted successfully."
3 }
```