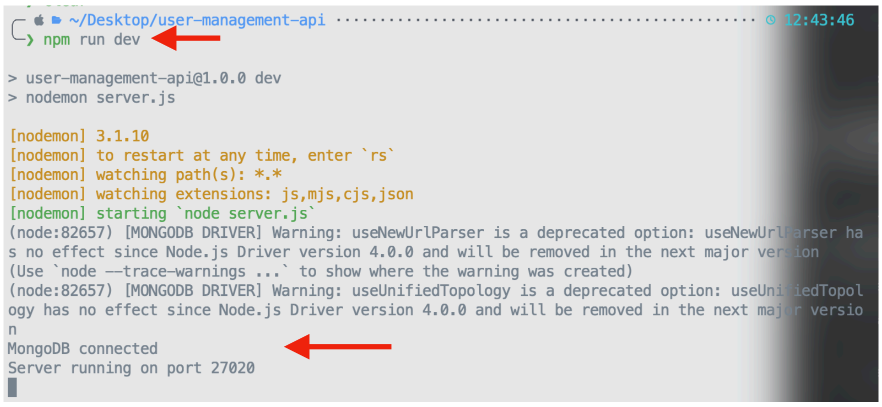
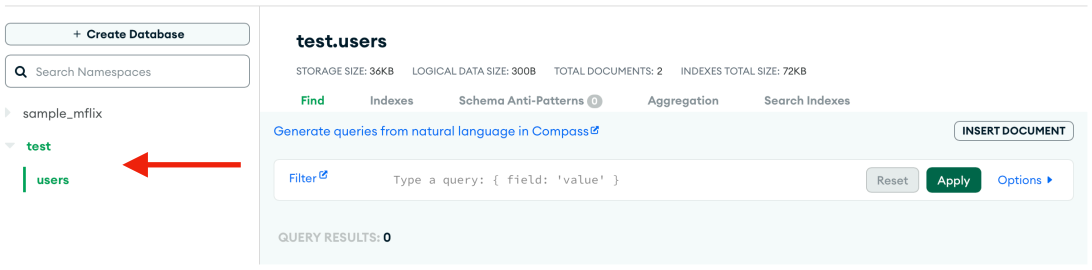
USER MANAGEMENT TESTING THROUGH POSTMAN

Connecting to MongoDB **[PORT 27020]** {**command : npm run dev**} **{run this command after you set your current directory to user-management-api}**



```
╭─  ▸ ~/Desktop/user-management-api ·········································· ◷ 12:43:46
╰─❯ npm run dev

> user-management-api@1.0.0 dev
> nodemon server.js

[nodemon] 3.1.10
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node server.js`
(node:82657) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser ha
s no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(Use `node --trace-warnings ...` to show where the warning was created)
(node:82657) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopol
ogy has no effect since Node.js Driver version 4.0.0 and will be removed in the next major versio
n
MongoDB connected
Server running on port 27020
```

<u>Initial Database - test: Collection - users</u>



STORAGE SIZE: 36KB   LOGICAL DATA SIZE: 300B   TOTAL DOCUMENTS: 2   INDEXES TOTAL SIZE: 72KB

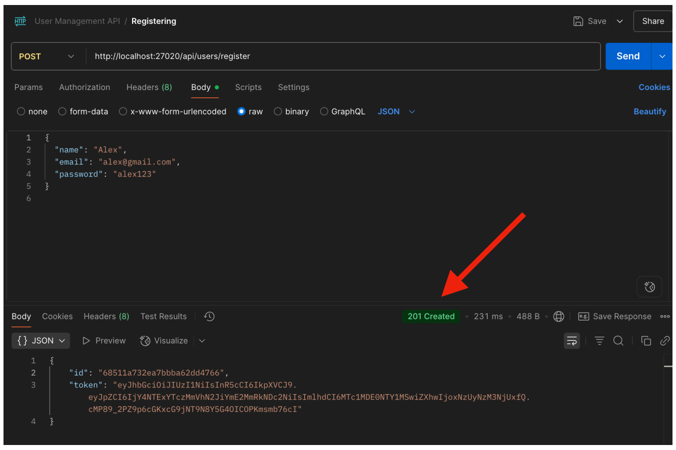
QUERY RESULTS: 0

1) REGISTERING A NEW USER [SENDING REQUEST USING POSTMAN]

Method : POST
URL : <u>http://localhost:27020/api/users/register</u>
Body : raw - json

```
{
 "name": "Alex",
 "email": “alex@gmail.com",
 "password": "alex123"
}
```

Sending request through Postman:



POST http://localhost:27020/api/users/register

Body: [✓ raw] JSON

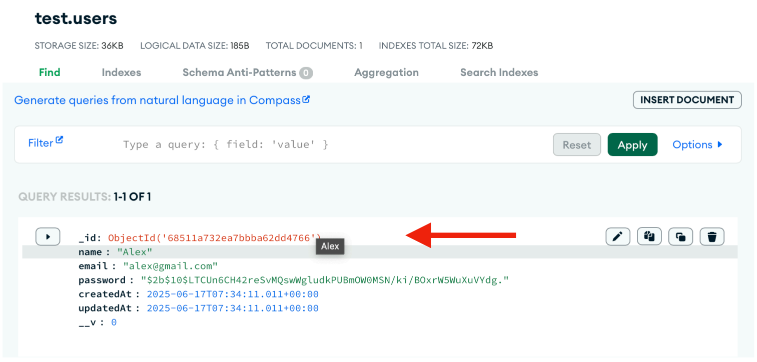
```
{
  "name": "Alex",
  "email": "alex@gmail.com",
  "password": "alex123"
}
```

201 Created · 231 ms · 488 B

```
{
    "id": "68511a732ea7bbba62dd4766",
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
        eyJpZCI6IjY4NTExYTczMmVhN2JiYmE2MmRkNDc2NiIsImlhdCI6MTc1MDE0NTY1MSwiZXhwIjoxNzUyNzM3NjUxfQ.
        cMP89_2PZ9p6cGKxcG9jNT9N8Y5G4OICOPKmsmb76cI"
}
```

<u>JWT Token :</u>
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY4NTExYTczMmVhN2JiYmE2MmRkNDc2NiIsImlhdCI6MTc1MDE0NTY1MSwiZXhwIjoxNzUyNzM3NjUxfQ.cMP89_2PZ9p6cGKxcG9jNT9N8Y5G4OICOPKmsmb76cI
[WE WILL NEED THIS FOR LATER REQUESTS]

User data successfuly sent to DB **[SUCCESSFUL REGISTRATION]**



test.users

STORAGE SIZE: 36KB   LOGICAL DATA SIZE: 185B   TOTAL DOCUMENTS: 1   INDEXES TOTAL SIZE: 72KB

QUERY RESULTS: 1-1 OF 1

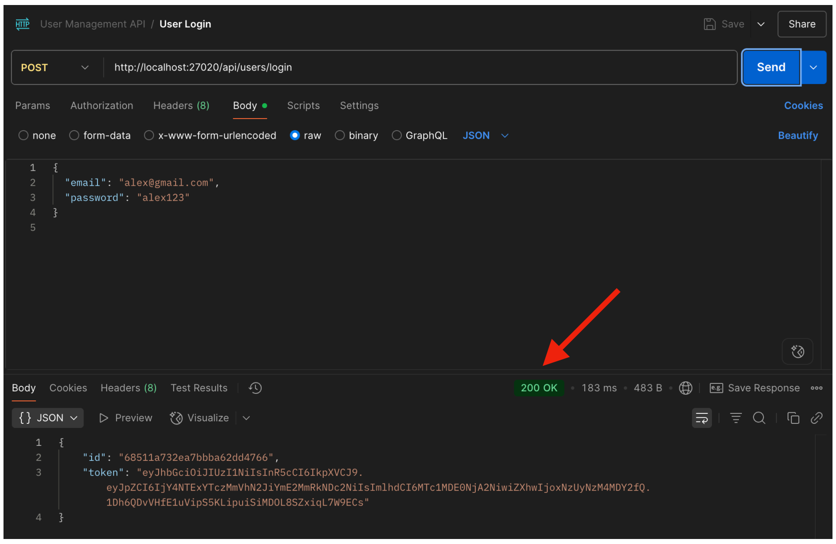
```
_id: ObjectId('68511a732ea7bbba62dd4766')
name : "Alex"
email : "alex@gmail.com"
password : "$2b$10$LTCUn6CH42reSvMQswWgludkPUBmOW0MSN/ki/BOxrW5WuXuVYdg."
createdAt : 2025-06-17T07:34:11.011+00:00
updatedAt : 2025-06-17T07:34:11.011+00:00
__v : 0
```

2) USER LOGIN

Method : POST
URL : http://localhost:27020/api/users/login
Body : raw - json

```
{
  "email": “alex@gmail.com",
  "password": “alex123"
}
```

Sending a login request through postmen **{200 OK : SUCCESSFULL }**



POST http://localhost:27020/api/users/login

Body: none form-data x-www-form-urlencoded [✓ raw] binary GraphQL JSON

```
{
  "email": "alex@gmail.com",
  "password": "alex123"
}
```

200 OK • 183 ms • 483 B

```
{
    "id": "68511a732ea7bbba62dd4766",
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
        eyJpZCI6IjY4NTExYTczMmVhN2JiYmE2MmRkNDc2NiIsImlhdCI6MTc1MDE0NjA2NiwiZXhwIjoxNzUyNzM4MDY2fQ.
        1Dh6QDvVHfE1uVipS5KLipuiSiMDOL8SZxiqL7W9ECs"
}
```

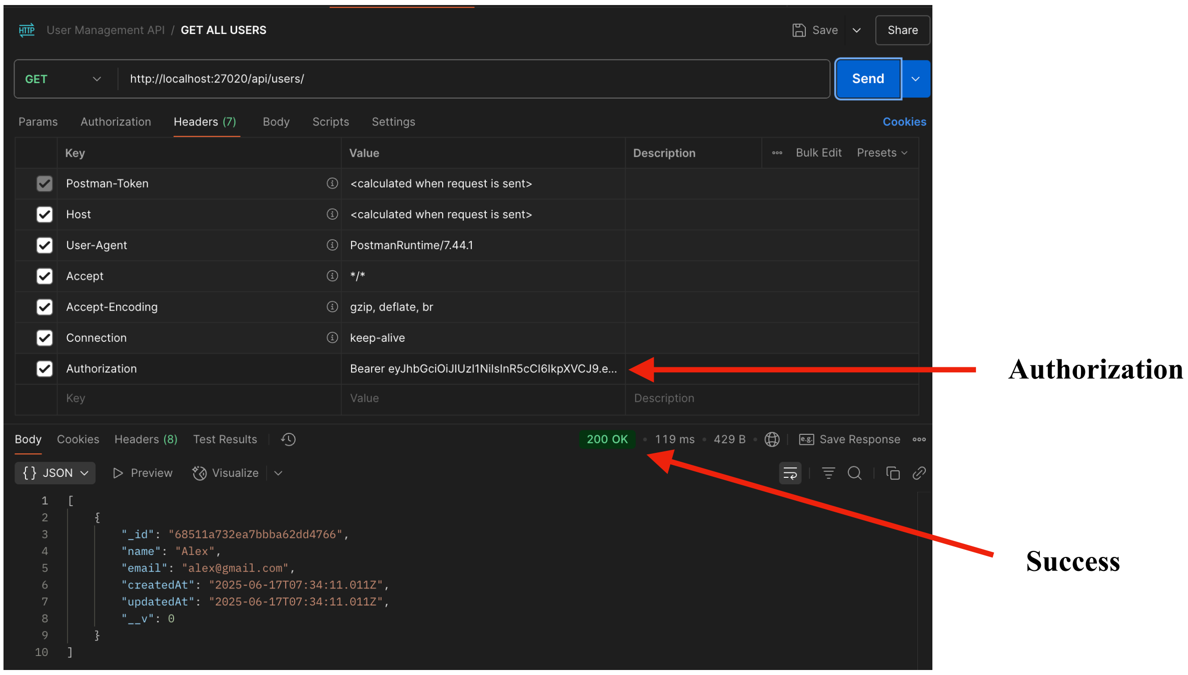
<u>JWT Token :</u>
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY4NTExYTczMmVhN2JiYmE2MmRkNDc2NiIsImlhdCI6MTc1MDE0NjA2NiwiZXhwIjoxNzUyNzM4MDY2fQ.1Dh6QDvVHfE1uVipS5KLipuiSiMDOL8SZxiqL7W9ECs **[WE WILL NEED THIS FOR LATER REQUESTS]**

3) GET ALL USERS (Protected Route - ADMIN ONLY)

Method : GET
URL : http://localhost:27020/api/users/
Body :
Headers : Key : Authorization, Value : Bearer <JWT_TOKEN>

Sending a 'GET ALL' Request to fetch existing users from database

We got our data of existing users (Only 1 user since only 1 user in database)



GET http://localhost:27020/api/users/

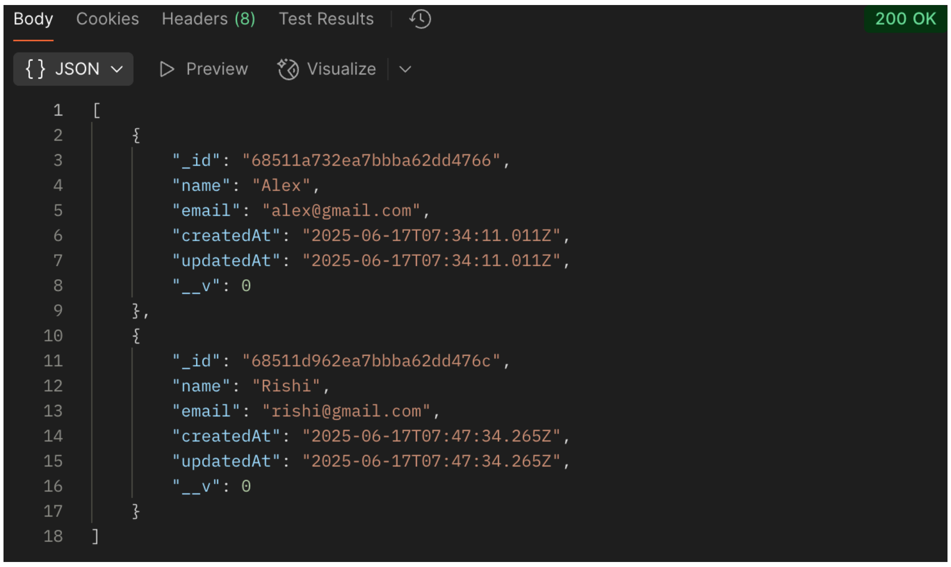
	Key	Value	Description
✓	Postman-Token	<calculated when request is sent>	
✓	Host	<calculated when request is sent>	
✓	User-Agent	PostmanRuntime/7.44.1	
✓	Accept	*/*	
✓	Accept-Encoding	gzip, deflate, br	
✓	Connection	keep-alive	
✓	Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e...	
	Key	Value	Description

200 OK • 119 ms • 429 B

```
[
    {
        "_id": "68511a732ea7bbba62dd4766",
        "name": "Alex",
        "email": "alex@gmail.com",
        "createdAt": "2025-06-17T07:34:11.011Z",
        "updatedAt": "2025-06-17T07:34:11.011Z",
        "__v": 0
    }
]
```

Registering a new user and trying the GET ALL request once again

I have registered a new user (Rishi, rishi@gmail.com, <password>)
Now when we send the request again we can see the data of existing users (2 users)



```
[
    {
        "_id": "68511a732ea7bbba62dd4766",
        "name": "Alex",
        "email": "alex@gmail.com",
        "createdAt": "2025-06-17T07:34:11.011Z",
        "updatedAt": "2025-06-17T07:34:11.011Z",
        "__v": 0
    },
    {
        "_id": "68511d962ea7bbba62dd476c",
        "name": "Rishi",
        "email": "rishi@gmail.com",
        "createdAt": "2025-06-17T07:47:34.265Z",
        "updatedAt": "2025-06-17T07:47:34.265Z",
        "__v": 0
    }
]
```

4) GETTING A SINGLE USER

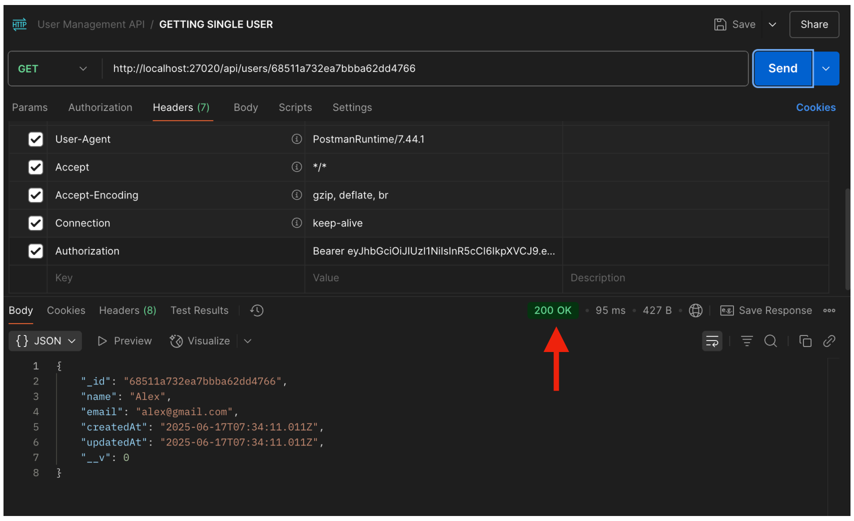
Method : GET
URL : http://localhost:27020/api/users/<user_id> **{replace <user_id> with id of the user which you are fetching}**

Replace <user_id> with actual ID from register/login

Headers : Key : Authorization, Value : Bearer <JWT_TOKEN>

Getting data of "alex" user (by using Id of alex : refer to 'registering a user' part to get the user_id)

Successfully fetched data of "alex"



GET http://localhost:27020/api/users/68511a732ea7bbba62dd4766

	Key	Value	Description
✓	User-Agent	PostmanRuntime/7.44.1	
✓	Accept	*/*	
✓	Accept-Encoding	gzip, deflate, br	
✓	Connection	keep-alive	
✓	Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e...	

```
{
    "_id": "68511a732ea7bbba62dd4766",
    "name": "Alex",
    "email": "alex@gmail.com",
    "createdAt": "2025-06-17T07:34:11.011Z",
    "updatedAt": "2025-06-17T07:34:11.011Z",
    "__v": 0
}
```

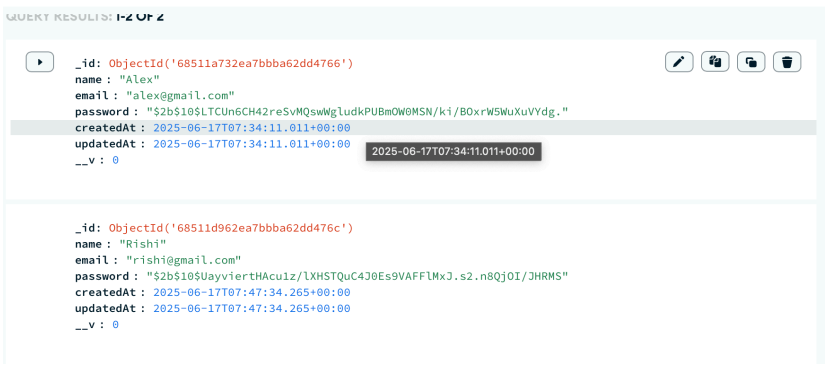
5) UPDATE A USER DETAILS (Protected Route - ADMIN ONLY)

Method : PUT
URL : http://localhost:27020/api/users/<user_id>
Headers : Authorization : Bearer <Token>

Body : raw - json

```
{
  "name": "Alice Updated",
  "email": "aliceUpdated@gmail.com"
}
```

DATABASE (BEFORE REQUEST):



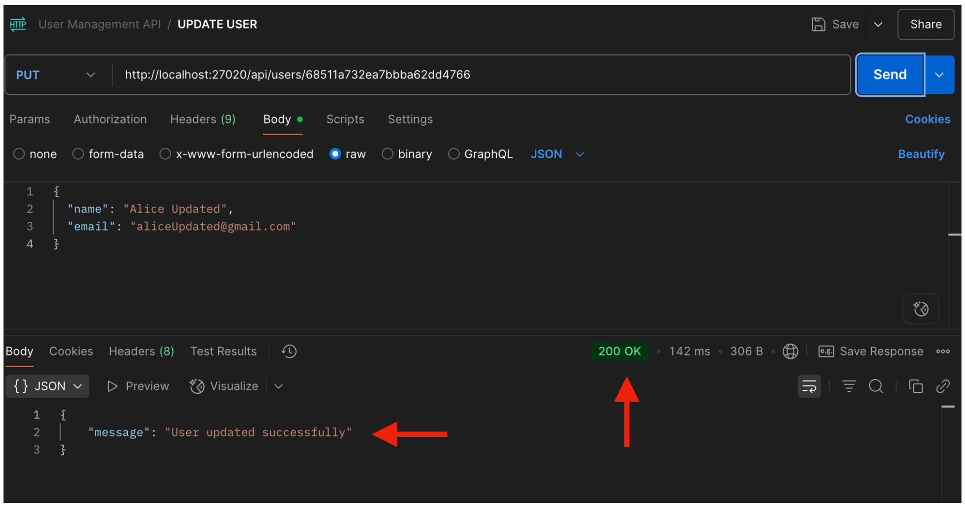
QUERY RESULTS: 1-2 OF 2

```
_id: ObjectId('68511a732ea7bbba62dd4766')
name : "Alex"
email : "alex@gmail.com"
password : "$2b$10$LTCUn6CH42reSvMQswWgludkPUBmOW0MSN/ki/BOxrW5WuXuVYdg."
createdAt : 2025-06-17T07:34:11.011+00:00
updatedAt : 2025-06-17T07:34:11.011+00:00
__v : 0
```

2025-06-17T07:34:11.011+00:00

```
_id: ObjectId('68511d962ea7bbba62dd476c')
name : "Rishi"
email : "rishi@gmail.com"
password : "$2b$10$UayviertHAcu1z/lXHSTQuC4J0Es9VAFFlMxJ.s2.n8QjOI/JHRMS"
createdAt : 2025-06-17T07:47:34.265+00:00
updatedAt : 2025-06-17T07:47:34.265+00:00
__v : 0
```

Sending a update request through postman



PUT http://localhost:27020/api/users/68511a732ea7bbba62dd4766

none form-data x-www-form-urlencoded [✓ raw] binary GraphQL JSON

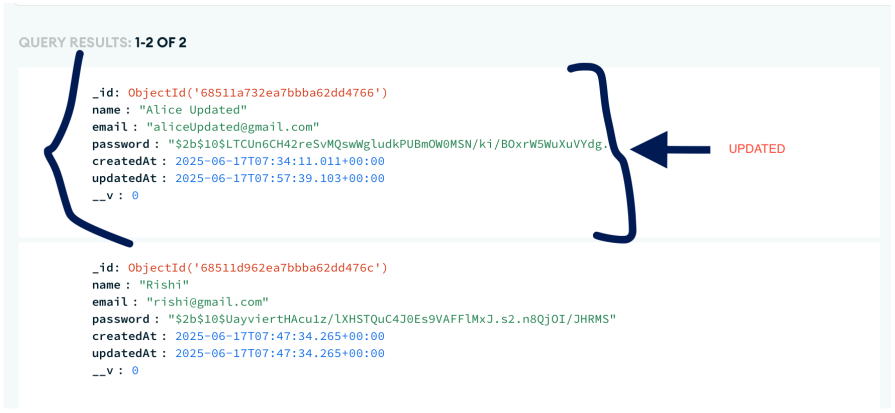
```
{
  "name": "Alice Updated",
  "email": "aliceUpdated@gmail.com"
}
```

200 OK • 142 ms • 306 B

```
{
    "message": "User updated successfully"
}
```

Request Successfull

DATABASE (AFTER REQUEST):



QUERY RESULTS: 1-2 OF 2

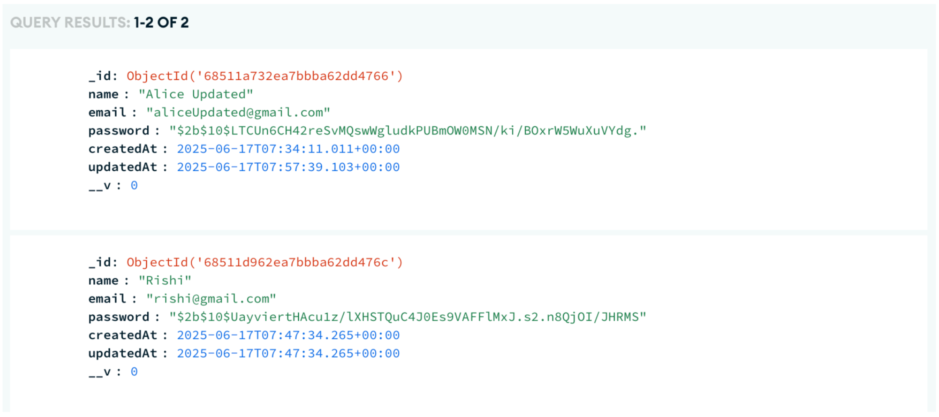
```
_id: ObjectId('68511a732ea7bbba62dd4766')
name : "Alice Updated"
email : "aliceUpdated@gmail.com"
password : "$2b$10$LTCUn6CH42reSvMQswWgludkPUBmOW0MSN/ki/BOxrW5WuXuVYdg."
createdAt : 2025-06-17T07:34:11.011+00:00
updatedAt : 2025-06-17T07:57:39.103+00:00
__v : 0

_id: ObjectId('68511d962ea7bbba62dd476c')
name : "Rishi"
email : "rishi@gmail.com"
password : "$2b$10$UayviertHAculz/lXHSTQuC4J0Es9VAFFlMxJ.s2.n8QjOI/JHRMS"
createdAt : 2025-06-17T07:47:34.265+00:00
updatedAt : 2025-06-17T07:47:34.265+00:00
__v : 0
```

6) DELETE A USER

Method : DELETE
URL : http://localhost:27020/api/users/<user_id>
Headers : Authorization : Bearer <Token>

In this case we are deleting the user "Rishi"

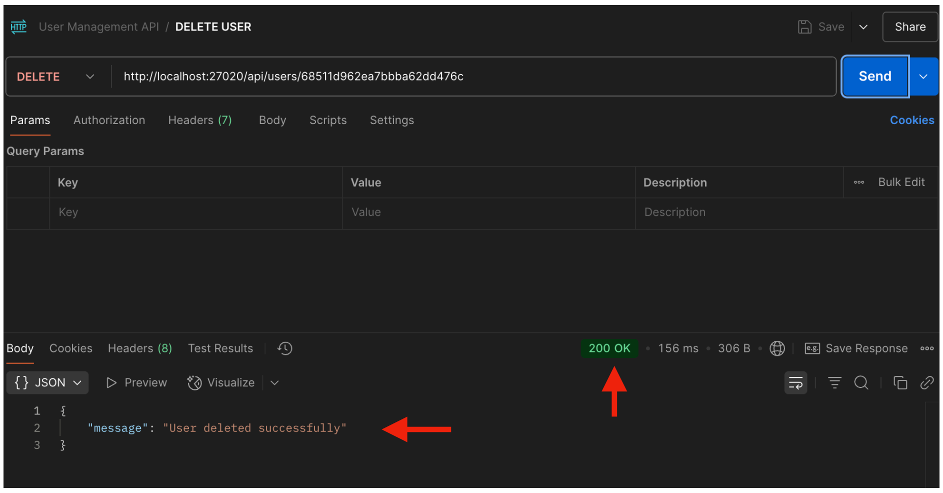
DATABASE (BEFORE REQUEST):



QUERY RESULTS: 1-2 OF 2

```
_id: ObjectId('68511a732ea7bbba62dd4766')
name : "Alice Updated"
email : "aliceUpdated@gmail.com"
password : "$2b$10$LTCUn6CH42reSvMQswWgludkPUBmOW0MSN/ki/BOxrW5WuXuVYdg."
createdAt : 2025-06-17T07:34:11.011+00:00
updatedAt : 2025-06-17T07:57:39.103+00:00
__v : 0

_id: ObjectId('68511d962ea7bbba62dd476c')
name : "Rishi"
email : "rishi@gmail.com"
password : "$2b$10$UayviertHAculz/lXHSTQuC4J0Es9VAFFlMxJ.s2.n8QjOI/JHRMS"
createdAt : 2025-06-17T07:47:34.265+00:00
updatedAt : 2025-06-17T07:47:34.265+00:00
__v : 0
```

Sending a delete user request through postman


User Management API / DELETE USER

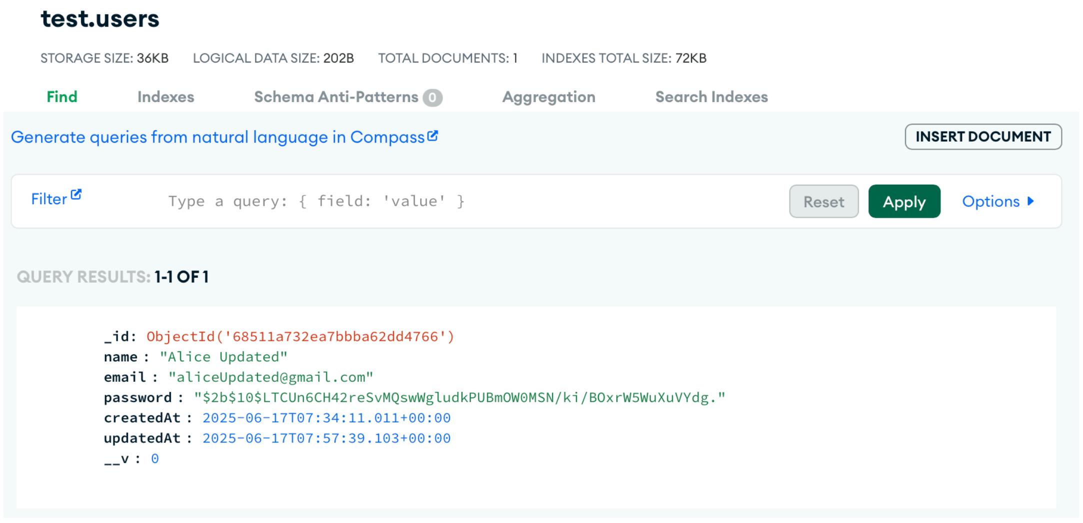
DELETE http://localhost:27020/api/users/68511d962ea7bbba62dd476c

Body 200 OK • 156 ms • 306 B

```
{
    "message": "User deleted successfully"
}
```

Successfull Request

<u>DATABASE (AFTER REQUEST):</u>


test.users

STORAGE SIZE: 36KB LOGICAL DATA SIZE: 202B TOTAL DOCUMENTS: 1 INDEXES TOTAL SIZE: 72KB

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('68511a732ea7bbba62dd4766')
name : "Alice Updated"
email : "aliceUpdated@gmail.com"
password : "$2b$10$LTCUn6CH42reSvMQswWgludkPUBmOW0MSN/ki/BOxrW5WuXuVYdg."
createdAt : 2025-06-17T07:34:11.011+00:00
updatedAt : 2025-06-17T07:57:39.103+00:00
__v : 0
```