

app.get("/products",fetchproducts);

The screenshot shows the Thunder Client interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** backendshoppyglobe
- Tab Bar:** THUNDER CLIENT, Shoppyglobe.model.js, Cart.model.js, shoppyRoutes.js, **localhost:8086/**, User.model.js, shoppyglobe.controller.js
- Left Sidebar:** Activity (selected), Collections, Env. It lists recent activity: GET localhost:8086/ just now, POST localhost:8081/ 22 hours ago, PUT localhost:5081/ 4 days ago, GET localhost:8080 5 days ago, POST localhost8888 1 month ago.
- Request Panel:** GET http://localhost:8086/products, Send button.
- Response Panel:** Status: 200 OK, Size: 8.49 KB, Time: 170 ms. Response tab shows the JSON data:

```
[{"_id": "69525067e37dbba64e21f120", "title": "Essence Mascara Lash Princess", "price": 9.99, "image": "https://cdn.dummyjson.com/product-images/1/1.jpg", "discount": 7.17, "brand": "Essence", "returnPolicy": "", "warrantyInformation": "", "description": "Volumizing and lengthening mascara.", "stock": 5, "minimumOrderQuantity": 24, "category": "beauty", "rating": "4.94", "discountPercentage": 7.17, "__v": 0}, {"_id": "69525067e37dbba64e21f121", "title": "iPhone X", "price": 899, "image": "https://cdn.dummyjson.com/product-images/2/1.jpg", "discount": 17.04}]
```
- Bottom Bar:** Response, Chart, Inline suggestions quota reached, Go Live, Quick Save, Top Stories, 10:14 AM, 12/30/2025

app.get("/products/:id",fetchproductbyid);

The screenshot shows a developer's environment with a code editor and a terminal. The code editor has tabs for Shoppyglobe.model.js, Cart.model.js, shoppyRoutes.js, localhost:8086/ (which is active), User.model.js, and shoppyglobe.controller.js. The terminal at the bottom has tabs for main* and 0△0. A search bar at the top says "backendshoppyglobe".

The main area displays a POST request to `http://localhost:8086/products/69525067e37dbba64e21f12`. The response status is 200 OK, size is 369 Bytes, and time is 108 ms. The response body is a JSON object representing a product:

```
1 {  
2   "_id": "69525067e37dbba64e21f12",  
3   "title": "Samsung Universe 9",  
4   "price": 1249,  
5   "image": "https://cdn.dummyjson.com/product-images/3/1.jpg",  
6   "discount": 15.46,  
7   "brand": "Samsung",  
8   "returnPolicy": "",  
9   "warrantyInformation": "",  
10  "description": "Premium Samsung smartphone.",  
11  "stock": 36,  
12  "minimumOrderQuantity": 1,  
13  "category": "smartphones",  
14  "rating": "4.75",  
15  "discountPercentage": 15.46,  
16  "__v": 0  
17 }
```

The left sidebar shows a list of recent activity:

- GET localhost:8086/ just now
- POST localhost:8081/ 22 hours ago
- PUT localhost:5081/ 4 days ago
- GET localhost:8080 5 days ago
- POST localhost:8888 1 month ago

The bottom taskbar includes icons for file operations, a search bar, and system notifications like "Inline suggestions quota reached" and "Afternoon rain". The system tray shows the date and time as 10:16 AM, 12/30/2025.

app.post("/register",verifyregisteruser,userRegister);

The screenshot shows the Postman application interface. The top navigation bar includes File, Edit, Selection, View, Go, Run, and a search bar for "backendshoppyglobe". Below the search bar are tabs for shopyRoutes.js, localhost:8086/, localhost:8081/, localhost/8080, localhost8888, User.model.js, and shopygc. The main area has tabs for Activity, Collections, Env, and filter activity. A "New Request" button is highlighted. The "Activity" tab is selected, showing a list of recent requests: POST localhost:8086/ (just now), POST localhost:8081/ (22 hours ago), PUT localhost:5081/ (4 days ago), GET localhost/8080 (5 days ago), and POST localhost8888 (1 month ago). A "POST http://localhost:8086/register" request is currently being viewed. The "Body" tab is selected, showing JSON content:

```
1 {  
2   "firstName": "MOHD",  
3   "middleName": "YASIR",  
4   "lastName": "taploo",  
5   "email": "yasir@gmail.com",  
6   "mobileNumber": "7006588550",  
7   "password": "Yasir@345"  
8 }
```

The "Response" tab shows the following JSON response:

```
1 {  
2   "message": "User registered successfully"  
3 }
```

Below the response, there are tabs for Headers, Cookies, Results, and Docs. The bottom right corner shows Response, Chart, and a dropdown menu. The taskbar at the bottom includes icons for File Explorer, Task View, Edge, File Manager, File Explorer, DEV C++, Chrome, and Paint. The system tray shows the date and time as 10:24 AM, 12/30/2025.

if we try to register the user from same email

The screenshot shows the Thunder Client interface with the following details:

- File Edit Selection View Go Run ...** - Top navigation bar.
- backendshoppyglobe** - Project name in the top bar.
- THUNDER CLIENT** - Left sidebar title.
- New Request** - Button in the top left of the main area.
- Activity** - Active tab in the sidebar.
- Collections** - Sidebar item.
- Env** - Sidebar item.
- filter activity** - Search bar in the sidebar.
- POST localhost:8086/** - Recent request listed in the sidebar, timestamped "1 min ago".
- POST localhost:8081/** - Recent request listed in the sidebar, timestamped "22 hours ago".
- PUT localhost:5081/** - Recent request listed in the sidebar, timestamped "4 days ago".
- GET localhost/8080** - Recent request listed in the sidebar, timestamped "5 days ago".
- POST localhost8888** - Recent request listed in the sidebar, timestamped "1 month ago".
- JS shoppyRoutes.js** - File tab in the top bar.
- localhost:8086/ X** - Request tab in the top bar.
- localhost:8081/** - Request tab in the top bar.
- localhost/8080** - Request tab in the top bar.
- localhost8888** - Request tab in the top bar.
- User.model.js** - File tab in the top bar.
- JS shoppyglobe** - File tab in the top bar.
- POST http://localhost:8086/register** - Current request being built.
- Send** - Send button for the current request.
- Query Headers³ Auth Body¹ Tests Pre Run** - Request builder tabs.
- Body¹** - Selected tab for the request body.
- JSON XML Text Form Form-encode GraphQL Binary** - Body format tabs.
- JSON Content** - Body content area.
- Format** - Body content area dropdown.
- 1 {
2 "firstName": "MOHD",
3 "middleName": "YASIR",
4 "lastName": "taploo",
5 "email": "yasir@gmail.com",
6 "mobileNumber": "7006588550",
7 "password": "Yasir@345"
8 }** - JSON content of the request body.
- Status: 409 Conflict Size: 34 Bytes Time: 118 ms** - Response summary.
- Response Headers⁶ Cookies Results Docs** - Response tabs.
- 1 {
2 "message": "email already exists"
3 }** - Response content.
- Copy** - Copy button for the response content.
- Response Chart** - Bottom navigation tabs.
- main* 0 0 0 0** - Bottom left status bar.
- Type here to search** - Bottom left search bar.
- Windows Start button** - Bottom left OS button.
- Cloud storage icons** - Bottom left cloud storage icons.
- Taskbar icons: File Explorer, Microsoft Edge, File Manager, Task View, Dev C++, Google Chrome, Paint 3D** - Bottom center taskbar icons.
- System tray icons: Battery, Network, Volume, ENG, 10:25 AM, 12/30/2025** - Bottom right system tray icons.
- Inline suggestions quota reached** - Bottom right status message.
- Go Live Quick Save** - Bottom right status buttons.

app.post("/login",userLogin);

The screenshot shows the Thunder Client application interface. On the left, there's a sidebar with various icons and a list of recent activity:

- POST localhost:8086/ just now
- POST localhost:8081/ 22 hours ago
- PUT localhost:5081/ 4 days ago
- GET localhost:8080 5 days ago
- POST localhost8888 1 month ago

The main area is focused on a new request:

New Request

Method: POST | URL: http://localhost:8086/login | Send

Body tab is selected. The JSON Content is:

```
1 {  
2   "email": "yasir@gmail.com",  
3   "password": "Yasir@345"  
4 }
```

Response tab shows the result:

Status: 200 OK | Size: 122 Bytes | Time: 248 ms

```
1 {  
2   "message": "welcome user",  
3   "token": "eyJhbGciOiJIUzI1NiJ9.eWFzaXJAZ21haWwuY29t  
4 .pUgwh7cn0nJjQ0LnQpOWg9G2pFRkK9z3U1_LhPoybFA"
```

Copy button is available for the response JSON.

app.post("/login",userLogin);

The screenshot shows the Thunder Client interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** backendshoppyglobe
- Tab Bar:** JS shoppyRoutes.js, POST localhost:8086/, POST localhost:8081/, GET localhost:8080, POST localhost:8888, JS User.model.js, JS shoppyg, ...
- Left Sidebar:** THUNDER CLIENT, New Request, Activity (selected), Collections, Env, filter activity, 1 unread notification.
- Request Panel:** Method: POST, URL: http://localhost:8086/login, Send button.
- Body Tab:** Status: 200 OK, Size: 122 Bytes, Time: 248 ms.
- Body Content:** Headers (3), Auth, Body (1), Tests, Pre Run. Format: JSON. Content: { "email": "yasar@gmail.com", "password": "Yasir@345" }.
- Response Tab:** Headers (6), Cookies, Results, Docs. Response content:

```
1 {
2   "message": "welcome user",
3   "token": "eyJhbGciOiJIUzI1NiJ9.eWFzaXJAZ21haWwuY29t
4     .pUgwh7cn0nJjQ0LnQpOWg9G2pFRkK9z3U1_LhPoybfA"
```
- Bottom Bar:** Response, Chart, main*, 0 △ 0, Type here to search, Dev Tools, VS Code, Chrome, Edge, 9°C Mostly cloudy, 10:28 AM, 12/30/2025, ENG, Quick Save, Go Live, Inline suggestions quota reached.

if i try wrong email

if i try wrong password

The screenshot shows the Thunder Client interface with the following details:

- File Edit Selection View Go Run ...** - Top navigation bar.
- Q backendshoppyglobe** - Search bar.
- THUNDER CLIENT** - Left sidebar tab.
- New Request** - Active tab in the left sidebar.
- Activity Collections Env** - Left sidebar tabs.
- filter activity** - Left sidebar search input.
- POST localhost:8086/ just now** - Recent activity log entry.
- POST localhost:8081/ 22 hours ago** - Recent activity log entry.
- PUT localhost:5081/ 4 days ago** - Recent activity log entry.
- GET localhost/8080 5 days ago** - Recent activity log entry.
- POST localhost8888 1 month ago** - Recent activity log entry.
- JS shoppyRoutes.js** - Tab in the top header.
- localhost:8086/ X** - Tab in the top header.
- localhost:8081/** - Tab in the top header.
- localhost/8080** - Tab in the top header.
- localhost8888** - Tab in the top header.
- User.model.js** - Tab in the top header.
- JS shoppyc ...** - Tab in the top header.
- POST http://localhost:8086/login** - Request URL in the main area.
- Send** - Send button.
- Status: 401 Unauthorized Size: 30 Bytes Time: 255 ms** - Response summary.
- Response Headers Cookies Results Docs** - Response tab navigation.
- Copy** - Copy button for the response content.
- JSON XML Text Form Form-encode GraphQL Binary** - Body format tabs.
- JSON Content** - Body content tab.
- Format** - Body format dropdown.
- 1 {
2 "email": "yasir@gmail.com",
3 "password": "Yas5"
4 }** - JSON body content.
- Response Chart** - Bottom navigation tabs.
- main* 0 0 0 0** - Bottom status bar.
- Type here to search** - Bottom search bar.
- File Explorer Task View Task List Dev C++ Chrome Paint** - Bottom taskbar icons.
- Inline suggestions quota reached Go Live Quick Save** - Bottom status bar.
- 9°C Mostly cloudy 10:30 AM 12/30/2025** - Bottom system tray.

after login i can add products

The screenshot shows the Thunder Client interface with the following details:

- Header Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** Q backendshoppyglobe
- Toolbar:** Back, Forward, Refresh, Stop, Minimize, Maximize, Close.
- Left Sidebar (Activity):** THUNDER CLIENT, New Request, Activity (selected), Collections, Env, filter activity. Recent requests: POST localhost:8086/ (just now), POST localhost:8081/ (22 hours ago), PUT localhost:5081/ (4 days ago), GET localhost:8080 (5 days ago), POST localhost:8888 (1 month ago).
- Request Section:** Method: POST, URL: http://localhost:8086/addproducts, Send button.
- Body Section:** Body tab selected, JSON tab selected, JSON Content: A JSON object representing a Samsung Galaxy S23 product.
- Response Section:** Status: 201 Created, Size: 498 Bytes, Time: 130 ms. Response tab selected, Headers (6), Cookies, Results, Docs. Response body: A JSON object indicating success and providing detailed product information.
- Bottom Status Bar:** Response, No Notifications, main*, 0 ▲ 0, Inline suggestions quota reached, Go Live, Quick Save, 9°C Mostly cloudy, 10:33 AM, 12/30/2025.

```
1 {
2   "success": true,
3   "message": "Product added successfully",
4   "data": [
5     {
6       "title": "Samsung Galaxy S23",
7       "price": 74999,
8       "image": "https://example.com/images/galaxy-s23.jpg",
9       "discount": 5000,
10      "brand": "Samsung",
11      "returnPolicy": "7 days replacement only",
12      "warrantyInformation": "1 year manufacturer warranty",
13      "description": "Samsung Galaxy S23 with 8GB RAM and 128GB storage.",
14      "stock": 50,
15      "minimumOrderQuantity": 1,
16      "category": "Electronics",
17      "rating": "4.5",
18      "discountPercentage": 6.6
19    }
20  ]
21 }
22 ]
23 }
```

after login i can delete products

The screenshot shows the Thunder Client interface with the following details:

- File Edit Selection View Go Run ...** - Top navigation bar.
- backendshoppyglobe** - Project name in the top bar.
- THUNDER CLIENT** - Left sidebar title.
- New Request** - New request button.
- Activity** (selected), Collections, Env - Activity tab selected.
- filter activity** - Filter input field.
- DEL localhost:8086/ just now** - Recent activity log entry.
- POST localhost:8081/ 22 hours ago**
- PUT localhost:5081/ 4 days ago**
- GET localhost:8080 5 days ago**
- POST localhost:8888 1 month ago**
- DELETE http://localhost:8086/products** - Request URL in the main panel.
- Send** - Send button.
- Status: 200 OK Size: 42 Bytes Time: 116 ms** - Response summary.
- Response Headers Cookies Results Docs** - Response tab selected.
- Copy** - Copy button for the response JSON.
- JSON Content Format** - JSON content area with the following code:

```
1 {  
2   "id": "69535e010d5e1dd867227511"  
3 }  
4
```
- Response Chart** - Bottom navigation tabs.
- main* 0 0 0** - Bottom left status bar.
- Type here to search** - Bottom left search bar.
- Windows taskbar icons: File Explorer, Edge, File Manager, Task View, DEV C++, Chrome, Paint 3D.**
- System tray icons: Cloud, 9°C Cloudy, Battery, Volume, ENG, 10:37 AM, 12/30/2025.**

after login i can access cart

The screenshot shows a Thunder Client interface with the following details:

- File Edit Selection View Go Run ...** - Top navigation bar.
- Q backendshoppyglobe** - Search bar.
- THUNDER CLIENT** - Left sidebar title.
- New Request** - A dropdown menu.
- Activity Collections Env** - Filter options.
- filter activity** - Input field for filtering activity.
- GET localhost:8086/ just now** - Recent activity log entry.
- POST localhost:8081/ 22 hours ago** - Recent activity log entry.
- PUT localhost:5081/ 4 days ago** - Recent activity log entry.
- GET localhost:8080 5 days ago** - Recent activity log entry.
- POST localhost:8888 1 month ago** - Recent activity log entry.
- localhost:8086/ X** - Active tab.
- localhost:8081/** - Tab.
- localhost:8080** - Tab.
- localhost:8888** - Tab.
- User.model.js** - Tab.
- shoppyglobe.controller.js M** - Tab.
- GET http://localhost:8086/cart Send** - Request configuration.
- Status: 404 Not Found Size: 30 Bytes Time: 122 ms** - Response summary.
- Response Headers Cookies Results Docs** - Response tabs.
- Copy** - Copy button for the response content.
- JSON Content** - JSON content pane showing the response body.
- Format** - Format dropdown in the JSON content pane.
- 1 { 2 "id": "69535e010d5e1dd867227511" 3 } 4** - The JSON response content.
- Response Chart** - Bottom navigation tabs.
- main* 0 0 0 0** - Bottom left status bar.
- Type here to search** - Bottom left search bar.
- File Explorer Taskbar Icons** - Bottom taskbar icons.
- Rain today 10:40 AM 12/30/2025** - Bottom right system tray.

after login i can add an item to the cart

The screenshot shows a development environment with multiple windows open. On the left, there's a sidebar with icons for file operations, a search bar, and a list of recent activity. The main window is a Thunder Client interface. It has tabs for localhost:8086, localhost:8081, localhost:8080, localhost:8888, User.model.js, and shop. A new request is being prepared for a POST to http://localhost:8086/cart. The Body tab is selected, showing JSON content:

```
1 {
2   "productid": "69536238564e8ad77f388834",
3   "productQuantity": "20"
4 }
```

The response status is 201 Created, size is 151 Bytes, and time is 231 ms. The response body is:

```
1 {
2   "cartitem": {
3     "productid": "69536238564e8ad77f388834",
4     "productQuantity": 20,
5     "_id": "695364846af4e6de6e7187c2",
6     "__v": 0
7   },
8   "message": "item added successfully"
9 }
```

On the right side of the screen, there's a terminal window showing Node.js logs. It includes error messages about syntax errors, node version information, nodemon logs, and MongoDB connection details.

```
SyntaxError: missing ) after argument list
at compileSourceTextModule
(node:internal/modules/esm/utils:344:16)
at ModuleLoader.moduleStrategy
(node:internal/modules/esm/translators:105:18)
at #translate
(node:internal/modules/esm/loader:534:12)
at ModuleLoader.loadAndTranslate
(node:internal/modules/esm/loader:581:27)

Node.js v22.15.0
[nodemon] app crashed - waiting for file changes before starting...
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Server is running on port 8086
Connected to MongoDB
yasin@gmail.com
{
  _id: new ObjectId('6953642930338b8d84fbfa58'),
  productid: new ObjectId('69536238564e8ad77f388834'),
  productQuantity: 20,
  __v: 0
}
yasin@gmail.com
null
```

At the bottom, there's a taskbar with various icons and system status indicators. The date and time are shown as 11:05 AM on 12/30/2025.

if i try to add the same item again in cart

The screenshot shows a Thunder Client interface with the following details:

- Activity Tab:** The "Activity" tab is selected, showing a list of previous requests.
- Request Tab:** A new request is being prepared to `http://localhost:8086/cart` via POST.
- Body Tab:** The body is set to JSON format, containing the following payload:

```
1 {
2   "productId": "69535f860d5e1dd867227515",
3   "productQuantity": "20"
4 }
```
- Status:** The response status is **400 Bad Request**, with a size of **37 Bytes** and a time of **67 ms**.
- Response Tab:** The response body is:

```
1 {
2   "message": "Product already in cart"
3 }
```
- Terminal:** On the right, a terminal window shows node.js logs, including messages about nodemon restarting due to changes and starting node index.js.

update quantity of cart item by id

The screenshot shows a development environment with a sidebar on the left containing file navigation, search, and activity logs. The main area features a request builder for a PUT operation to update a cart item. The request URL is `http://localhost:8086/cart/69536238564e8ad77f388834`. The Body tab contains the following JSON payload:

```
1 {
2   "id": "69536238564e8ad77f388834",
3   "quantity": "80"
4 }
```

The Response section shows a successful 200 OK status with a size of 26 bytes and a response time of 184 ms. The response body is:

```
1 {
2   "message": "item updated"
3 }
```

On the right side, a terminal window displays the Node.js nodemon process restarting due to changes.

Activity Log (Left Sidebar):

- PUT localhost:8086/ just now
- POST localhost:8081/ 23 hours ago
- PUT localhost:5081/ 4 days ago
- GET localhost:8080 5 days ago
- POST localhost:8888 1 month ago

Request Builder (Main Area):

- Method: PUT
- URL: `http://localhost:8086/cart/69536238564e8ad77f388834`
- Body Tab selected
- JSON Content:

```
1 {
2   "id": "69536238564e8ad77f388834",
3   "quantity": "80"
4 }
```
- Status: 200 OK
- Size: 26 Bytes
- Time: 184 ms
- Response Tab selected
- Response Body:

```
1 {
2   "message": "item updated"
3 }
```

Terminal (Right Side):

```
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
```

System Tray (Bottom Right):

- Windows Start button
- Type here to search
- File Explorer icon
- Dev C++ icon
- Google Chrome icon
- Cloudy weather icon: 10°C
- Network icon
- Volume icon
- Language: ENG
- Date: 12/30/2025

if wrong id was sent then it will show the message accordingly

The screenshot shows a Thunder Client interface with the following details:

- Activity:** PUT localhost:8086/cart
- Body:** JSON content: { "id": "69536238564e8ad77f388824", "quantity": "90" }
- Status:** 404 Not Found
- Response:** { "message": "Cart item not found" }
- Logs:** Shows multiple log entries from nodemon indicating restarts due to changes and starting node index.js.

Thunder Client interface elements include:

- File, Edit, Selection, View, Go, Run, ...
- Search bar: backendshoppyglobe
- Tab bar: shopyRoutes.js, localhost:8086/, localhost:8081/, localhost:8080, localhost:8888, ...
- New Request button
- Activity, Collections, Env tabs
- filter activity search bar
- Recent requests list: PUT localhost:8086/ just now, POST localhost:8081/ 23 hours ago, PUT localhost:5081/ 4 days ago, GET localhost:8080 5 days ago, POST localhost:8888 1 month ago
- Response tab: Headers, Cookies, Results, Docs
- System tray icons: Windows, search, file explorer, task manager, VS Code, DEV C++, browser, calendar, weather (10°C Cloudy), network, battery, ENG, 11:59 AM, 12/30/2025

ORGANIZATION
Yasir's Org - 2025-11-19PROJECT
shoppyglobetwo**Data Explorer**

My Queries

Data Modeling

CLUSTERS (1)

Search clusters

Cluster0

admin

local

test

carts

products

users

Cluster0 > test

View monitoring

Visualize your data

Create collection

Refresh

Collection name	Properties	Storage size	Documents	Avg. document size	Indexes	Total index size
carts	-	32.77 kB	0	0 B	1	32.77 kB
products	-	36.86 kB	27	368.00 B	1	36.86 kB
users	-	36.86 kB	1	225.00 B	1	36.86 kB

three collections created

System Status: All Good

@2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Type here to search



10°C Cloudy

ENG
12/30/2025

ORGANIZATION
Yasir's Org - 2025-11-19PROJECT
shoppyglobetwo**Data Explorer**

{ } My Queries

Data Modeling

CLUSTERS (1)

Search clusters

Cluster0

admin

local

test

carts

products

users

carts

test

+

Cluster0 > test > carts

View monitoring

Visualize Your Data

Documents 0

Aggregations

Schema

Indexes 1

Validation

Type a query: { field: 'value' } or [Generate query](#) +

Explain

Reset

Find

</>

Options ▾

+ ADD DATA

UPDATE

DELETE

25 1 - 1 of 1

↻

<

>

▼

{}

田

```
_id: ObjectId('69537dee4bc9f72491ed84b6')
productid : ObjectId('69536238564e8ad77fd38894')
productQuantity : 1
__v : 0
```

cart collection

System Status: All Good

©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales



IST Internshala Trainings Project: Build APIs with Node Cluster0 Data | Cloud: MongoDB New Tab

cloud.mongodb.com/v2/69524271e41bf67355c9b5bf#/explorer/6952428f45800142cc175000/test/products/find

All Bookmarks

ORGANIZATION Yasir's Org - 2025-11-19 PROJECT shoppyglobetwo

Data Explorer carts products test +

Cluster0 > test > products

Documents 27 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#).

Explain Reset Find Options

Practice creating, reading, updating, and deleting documents efficiently. [GO TO SKILLS](#)

ADD DATA UPDATE DELETE

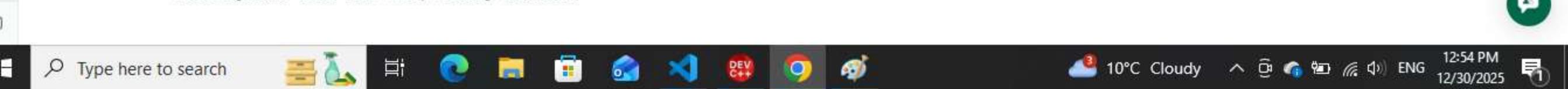
25 1 - 25 of 27

product collection

```
_id: ObjectId('69525067e37dbba64e21f128')
title: "Essence Mascara Lash Princess"
price: 9.99
image: "https://cdn.dummyjson.com/product-images/1/1.jpg"
discount: 7.17
brand: "Essence"
returnPolicy: ""
warrantyInformation: ""
description: "Volumizing and lengthening mascara."
stock: 5
minimumOrderQuantity: 24
category: "beauty"
rating: "4.94"
discountPercentage: 7.17
__v: 0
```

System Status: All Good

©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales



IST Internshala Trainings Project: Build APIs with Node Cluster0 Data | Cloud: MongoDB New Tab

cloud.mongodb.com/v2/69524271e41bf67355c9b5bf#/explorer/6952428f45800142cc175000/test/users/find

All Bookmarks

ORGANIZATION
Yasir's Org - 2025-11-19 PROJECT
shoppyglobetwo

Data Explorer

My Queries View monitoring Visualize Your Data

Data Modeling

CLUSTERS (1)

Search clusters

Cluster0

- admin
- local
- test
 - carts
 - products
 - users

Documents 1 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) + Explain Reset Find Options

Practice creating, reading, updating, and deleting documents efficiently. [GO TO SKILLS](#)

+ ADD DATA UPDATE DELETE

25 1-1 of 1

```
_id: ObjectId('69535b881440bb0bfd54f851')
firstName : "MOHD"
middleName : "YASIR"
lastName : "taploo"
email : "yasar@gmail.com"
mobileNumber : "7886588558"
password : "$2b$10$IwEl/yp2kgsQ.FjBtk4bKOInajjILofk6muWpG03ttd.4IBTk98jy"
__v : 0
```

user collection with registered users

System Status: All Good

©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Type here to search

10°C Cloudy 12:55 PM 12/30/2025