

## **EXPERIMENT NO. 6 - MongoDB**

Name of Student	Anuprita Mhapankar
Class Roll No	28
D.O.P.	13/03/2025
D.O.S.	20/03/2025
Sign and Grade	

**AIM:** To study CRUD operations in MongoDB

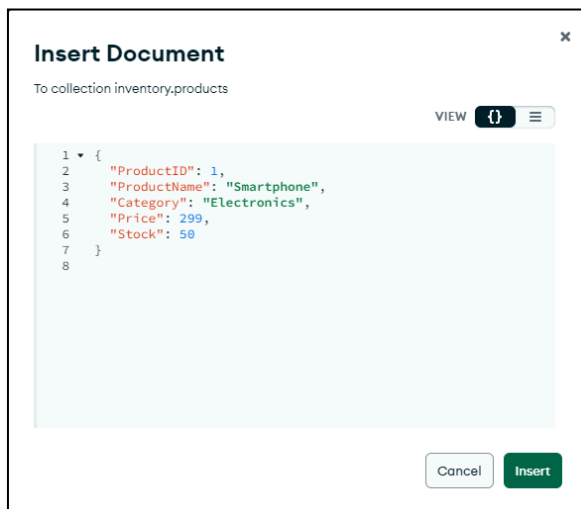
### **OVERVIEW OF TASKS PERFORMED :**

The experiment involves working with MongoDB to create and manipulate a database. The **"inventory"** database and **"products"** collection were created with fields such as ProductID, ProductName, Category, Price, and Stock. Ten documents were inserted, followed by various queries to retrieve all products, filter by category, sort by name, limit results, count products, hide the `_id` field, find distinct categories, and apply range conditions. Additionally, a product's price was updated, and a specific product entry was deleted.

**GITHUB LINK** - [https://github.com/Anuprita2022-26/WebX\\_Exp6](https://github.com/Anuprita2022-26/WebX_Exp6)

### **OUTPUT :**

#### **Insert Data (Create Operation)**



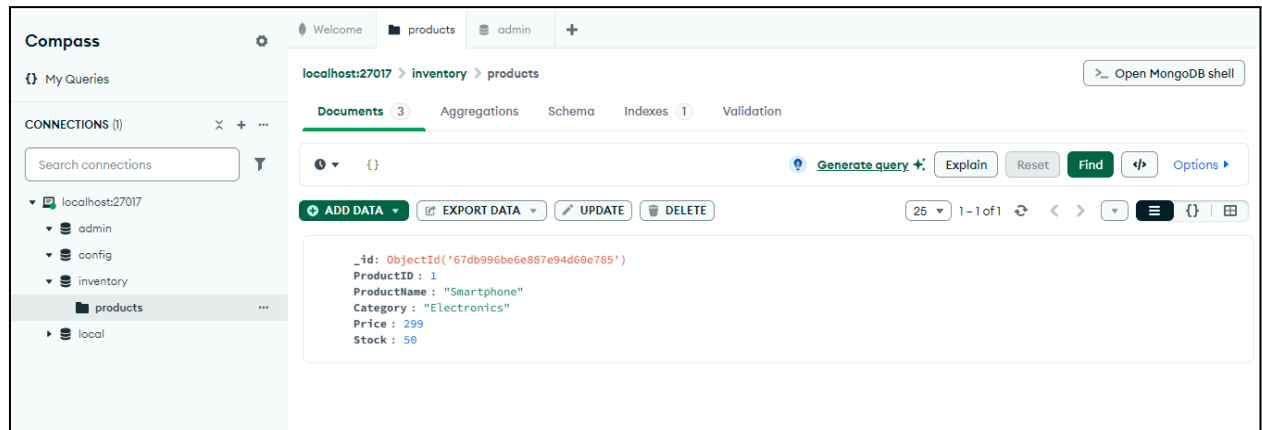
**Insert Document** x

To collection inventory.products

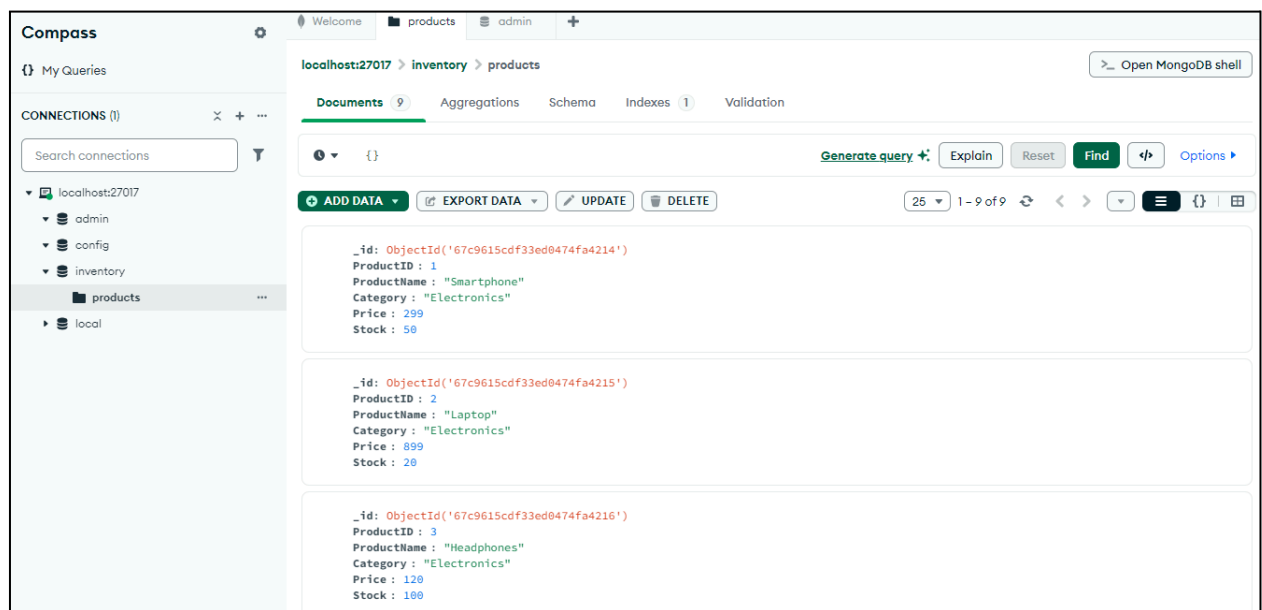
VIEW **JSON** ≡

```
1 {
2   "ProductID": 1,
3   "ProductName": "Smartphone",
4   "Category": "Electronics",
5   "Price": 299,
6   "Stock": 58
7 }
8
```

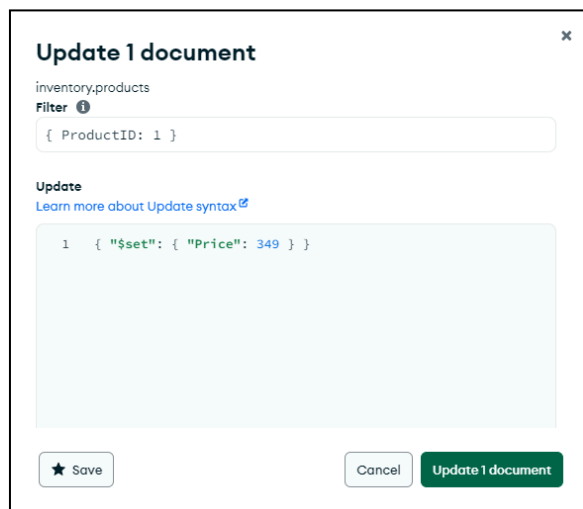
Cancel Insert

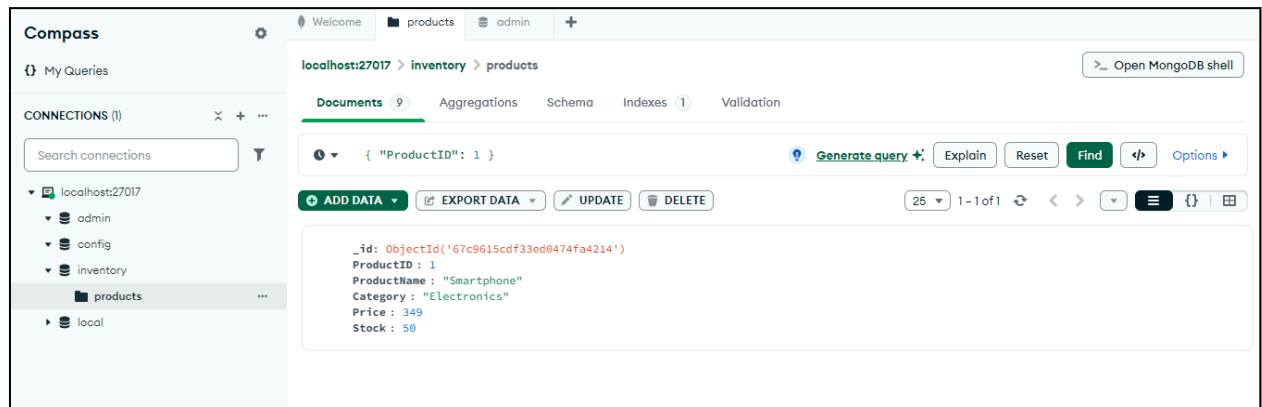


## Read Data (Retrieve Documents)

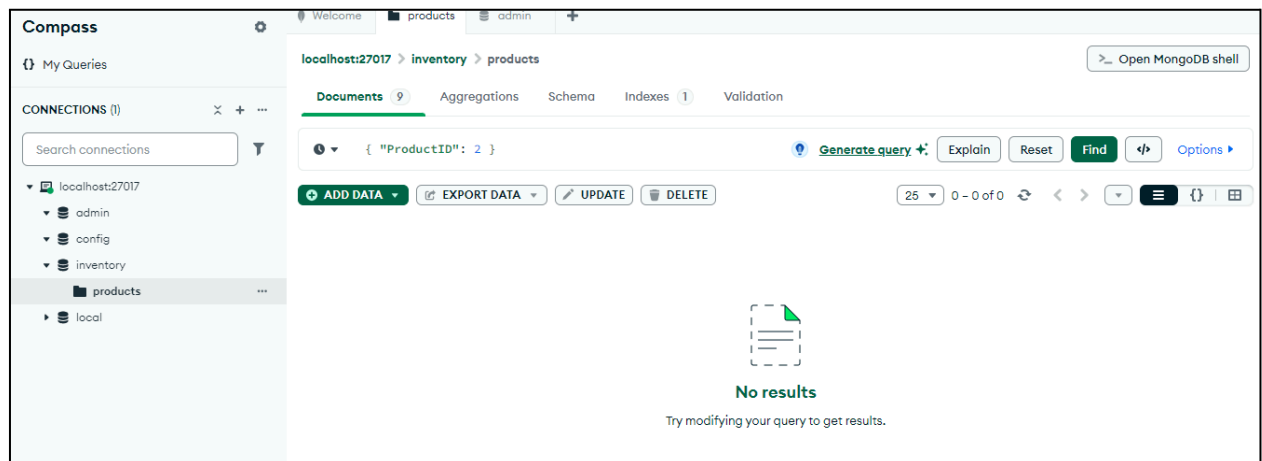
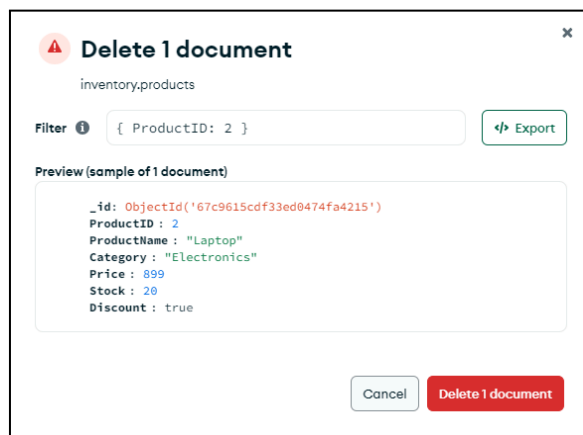


## Update Data





## Delete Data



## CONCLUSION

Through this experiment, we successfully performed **CRUD operations in MongoDB**, including **creating a database**, **inserting documents**, **querying data**, **updating records**, and **deleting entries**. We also explored filtering data, sorting, and aggregation queries.

MongoDB's document-oriented structure and flexible schema make it an ideal choice for handling large-scale, unstructured data in real-world applications.