CRUD Operations in MongoDB

Imagine you are tasked with managing the product catalog for an e-commerce platform. Products in the catalog have varying attributes, such as name, price, category, and inventory count. Over time, new product features like discounts and reviews need to be added. You must handle all of this data using MongoDB's flexible schema.

Activity Steps

1) Database and Collection Setup:

- Create a new MongoDB database called ecommerce.
- Define a collection called products to store product data.

2) Insert Product Data:

- Insert documents into the products collection representing different types of products (e.g., electronics, clothing, furniture).
- Each product should have a dynamic structure, meaning some products may have fields like discounts or reviews while others may not.

Example python code:

```
products.insert_many([
    { "name": "Laptop", "price": 1200, "category": "Electronics",
   "inventory": 10 },
    { "name": "T-Shirt", "price": 20, "category": "Clothing", "
   inventory": 50 },
    { "name": "Sofa", "price": 700, "category": "Furniture", "
   inventory": 5, "reviews": [{"user": "Alice", "rating": 4}] }
])
```

3) Perform CRUD Operations:

- Create: Insert at least 5 more products with varying attributes.
- Read: Query all products in the Electronics category where the price is greater than 500.
- Update: Add a discount field to all Furniture products and update the price accordingly.
- Delete: Remove products with inventory counts less than 5.

4) Schema Evolution:

- Add new fields (e.g., reviews or ratings) to some documents to simulate evolving product features over time. Explore MongoDB's capability to handle changing schemas dynamically.