MongoDB Case Study – Online Retail Product Catalog System

## 1. Product Catalog

MongoDB Schema Design:

{  
 \_id: ObjectId(),  
 name: "Wireless Mouse",  
 category: "Electronics",  
 price: 799,  
 stock: 150,  
 specifications: {  
 brand: "Logitech",  
 color: "Black",  
 connectivity: "Wireless",  
 battery\_life: "12 months"  
 }  
}

Another Sample Product:

{  
 \_id: ObjectId(),  
 name: "Running Shoes",  
 category: "Footwear",  
 price: 2999,  
 stock: 75,  
 specifications: {  
 brand: "Nike",  
 size: [7, 8, 9, 10],  
 color: "Blue",  
 material: "Mesh"  
 }  
}

## 2. Customer Orders

MongoDB Schema Design:

{  
 \_id: ObjectId(),  
 user\_id: ObjectId("60b8d295f1d3a2c3f8f264c9"),  
 order\_date: ISODate("2025-08-01T10:30:00Z"),  
 products: [  
 {  
 product\_id: ObjectId("60b8d2e2f1d3a2c3f8f264ca"),  
 name: "Wireless Mouse",  
 quantity: 2,  
 price: 799  
 },  
 {  
 product\_id: ObjectId("60b8d2f7f1d3a2c3f8f264cb"),  
 name: "Keyboard",  
 quantity: 1,  
 price: 1199  
 }  
 ],  
 total\_cost: 2797,  
 status: "Shipped"  
}

## 3. User Authentication

MongoDB Schema Design:

{  
 \_id: ObjectId(),  
 username: "john\_doe",  
 email: "john@example.com",  
 password\_hash: "5f4dcc3b5aa765d61d8327deb882cf99", // example hash  
 created\_at: ISODate("2025-07-15T14:00:00Z"),  
 last\_login: ISODate("2025-08-01T09:15:00Z")  
}

## 4. Querying and Indexing

1. Retrieve all products in the 'Electronics' category:

db.products.find({ category: "Electronics" })

2. Find all orders placed by a specific user:

db.orders.find({ user\_id: ObjectId("60b8d295f1d3a2c3f8f264c9") })

3. Authenticate a user by email and password hash:

db.users.findOne({ email: "john@example.com", password\_hash: "5f4dcc3b5aa765d61d8327deb882cf99" })