MongoDB Aggregation `$lookup` – Interview & Real-Project Oriented Guide

---

1. Collections Overview

We use **3 collections**:

- `users` → who places orders
- `products` → what is ordered
- `orders` → transactional data

Relationship:

```
users (1) ———< orders >——— (1) products
```

---

2. Insert Sample Data (INTERVIEW-LEVEL)

2.1 Users (10 Documents)

```
db.users.insertMany([
  { _id: 1, name: "Alice", city: "NY" },
  { _id: 2, name: "Bob", city: "LA" },
  { _id: 3, name: "Charlie", city: "TX" },
  { _id: 4, name: "David", city: "NY" },
  { _id: 5, name: "Eva", city: "LA" },
  { _id: 6, name: "Frank", city: "TX" },
  { _id: 7, name: "Grace", city: "NY" },
  { _id: 8, name: "Henry", city: "LA" },
  { _id: 9, name: "Ivy", city: "TX" },
  { _id: 10, name: "Jack", city: "NY" }
])
```

2.2 Products (10 Documents)

```
db.products.insertMany([
  { _id: 101, name: "Laptop", price: 1200 },
  { _id: 102, name: "Phone", price: 800 },
  { _id: 103, name: "Tablet", price: 600 },
  { _id: 104, name: "Headphones", price: 150 },
  { _id: 105, name: "Keyboard", price: 100 },
```

```
  { _id: 106, name: "Mouse", price: 50 },
  { _id: 107, name: "Monitor", price: 300 },
  { _id: 108, name: "Printer", price: 250 },
  { _id: 109, name: "Camera", price: 900 },
  { _id: 110, name: "Speaker", price: 200 }
])
```

2.3 Orders (Multiple Months & Years)

```
db.orders.insertMany([
  { _id: 1, userId: 1, productId: 101, quantity: 1, orderDate:
ISODate("2024-01-10") },
  { _id: 2, userId: 1, productId: 102, quantity: 2, orderDate:
ISODate("2024-01-20") },
  { _id: 3, userId: 2, productId: 103, quantity: 1, orderDate:
ISODate("2024-02-05") },
  { _id: 4, userId: 2, productId: 104, quantity: 3, orderDate:
ISODate("2024-02-18") },
  { _id: 5, userId: 3, productId: 105, quantity: 1, orderDate:
ISODate("2024-03-01") },
  { _id: 6, userId: 3, productId: 106, quantity: 5, orderDate:
ISODate("2024-03-12") },
  { _id: 7, userId: 4, productId: 107, quantity: 2, orderDate:
ISODate("2025-01-08") },
  { _id: 8, userId: 4, productId: 108, quantity: 1, orderDate:
ISODate("2025-01-22") },
  { _id: 9, userId: 4, productId: 109, quantity: 1, orderDate:
ISODate("2025-01-25") },
  { _id: 10, userId: 5, productId: 110, quantity: 4, orderDate:
ISODate("2025-02-14") }
])
```

---

3. Basic `$lookup` (User → Orders)

Interview Question: Get all users with their orders

```
db.users.aggregate([
  {
    $lookup: {
      from: "orders",
      localField: "_id",
      foreignField: "userId",
      as: "orders"
    }
```

```
  }
])
```

---

4. Join 3 Collections (User → Orders → Products)

Interview Question: Show user name, product name, price, quantity

```
db.users.aggregate([
  { $lookup: { from: "orders", localField: "_id", foreignField: "userId",
as: "orders" } },
  { $unwind: "$orders" },
  { $lookup: { from: "products", localField: "orders.productId",
foreignField: "_id", as: "product" } },
  { $unwind: "$product" },
  {
    $project: {
      _id: 0,
      user: "$name",
      product: "$product.name",
      price: "$product.price",
      quantity: "$orders.quantity",
      orderDate: "$orders.orderDate"
    }
  }
])
```

---

5. Who Made Maximum Orders in a Single Month

Interview Favorite Question

```
db.orders.aggregate([
  {
    $group: {
      _id: {
        userId: "$userId",
        month: { $month: "$orderDate" },
        year: { $year: "$orderDate" }
      },
      totalOrders: { $sum: 1 }
    }
  },
  { $sort: { totalOrders: -1 } },
  { $limit: 1 },
  { $lookup: { from: "users", localField: "_id.userId", foreignField: "_id",
as: "user" } },
```

```
  { $unwind: "$user" }
])
```

---

6. Who Made Maximum Orders in a Year

```
db.orders.aggregate([
  {
    $group: {
      _id: { userId: "$userId", year: { $year: "$orderDate" } },
      totalOrders: { $sum: 1 }
    }
  },
  { $sort: { totalOrders: -1 } },
  { $limit: 1 },
  { $lookup: { from: "users", localField: "_id.userId", foreignField: "_id",
as: "user" } },
  { $unwind: "$user" }
])
```

---

7. Total Amount Spent by Each User

```
db.users.aggregate([
  { $lookup: { from: "orders", localField: "_id", foreignField: "userId",
as: "orders" } },
  { $unwind: "$orders" },
  { $lookup: { from: "products", localField: "orders.productId",
foreignField: "_id", as: "product" } },
  { $unwind: "$product" },
  {
    $group: {
      _id: "$name",
      totalSpent: { $sum: { $multiply: ["$orders.quantity",
"$product.price"] } }
    }
  }
])
```

---

8. Most Sold Product (Quantity Wise)

```
db.orders.aggregate([
  {
    $group: {
```

```
      _id: "$productId",
      totalQuantity: { $sum: "$quantity" }
    }
  },
  { $sort: { totalQuantity: -1 } },
  { $limit: 1 },
  { $lookup: { from: "products", localField: "_id", foreignField: "_id", as:
"product" } },
  { $unwind: "$product" }
])
```

---

9. Common Interview Questions You Can Now Answer

- Who placed the most orders?
- Monthly / yearly analytics
- User purchase history
- Best selling product
- Total revenue per user

🟢BASIC LEVEL (Understand JOIN Concept)

1. Get all users along with their orders.

2. Get all orders with user name instead of userId.

3. Get all orders with product name and price.

4. Show user name and total number of orders placed.

5. Find users who have NOT placed any orders.

🟡INTERMEDIATE LEVEL (Real Interview Favorites)

6. Show user name, product name, quantity, and order date (3 collection join).

7. Find total quantity ordered by each user.

8. Find total amount spent by each user.

9. Find all orders placed in January 2025 with user name.

10. Find the most sold product (by quantity).

🟠INTERMEDIATE → ADVANCED (Analytics Based)

11. Find who placed the maximum number of orders in a single month.

12. Find who placed the maximum number of orders in a year.

13. Find monthly order count for each user.

14. Find yearly total revenue.

15. Find users who ordered more than 2 different products.

🔴ADVANCED LEVEL (INTERVIEW WINNERS)

16. Find the top 3 users based on total spending.

17. Find the least sold product.

18. Find users who ordered only once.

19. Find products that were never ordered.

20. For each user, show:

- total orders
- total quantity
- total amount spent

(single aggregation output)

---

10. How to Practice (VERY IMPORTANT)

For each question:

1. Think which collection to start with
2. Decide where `$lookup` is needed
3. Use `$unwind` only when required
4. Finish with `$group` or `$project`

---

11. Interview Tip (Memorize This Explanation)

> "\$lookup `is used when data is normalized across collections. In real projects, we mostly use it for reporting and analytics like dashboards, monthly summaries, and admin panels. We usually combine` \$lookup `with` \$unwind `,` \$group `, and` \$project` to get meaningful insights."