# MongoDB Compass Import + Analysis Tasks

MongoDB Compass Import Instructions (Tested November 2025)

1) Open MongoDB Compass and connect to your cluster (Atlas or local).
2) Click “+ Create Database” → Database: demo, Collection: orders (or use any names you prefer).
3) Inside the new collection, click “ADD DATA” → “Import File”.
4) Choose File: orders_compass.json (provided with this package).
5) Select File Type: JSON.
6) Import Method: “JSON array” (since the file contains an array of documents).
7) Click “Import”. You should see ~150 documents.
8) (Optional) Create helpful indexes for performance:
   - {"order_date": 1}
   - {"customer.customer_id": 1}
   - {"shipping.address.city": 1}
   - {"items.category": 1}
   - {"shipping.geo": "2dsphere"}
9) (Optional) If you want ISODate types instead of strings for order_date/delivered_at, run an update in Compass “Aggregations” or “Playground”:

```
db.orders.updateMany(
  { },
  [
    { $set: {
        order_date: { toDate:order_date" },
        "shipping.delivered_at": {
          $cond: [{ $ifNull: ["$shipping.delivered_at", false] },
                  { toDate:shipping.delivered_at" },
                  null]
        }
    }}
  ]
)
```

MongoDB Analysis & Query Tasks (15)

Q1. Total revenue (INR) by month (use charges.grand_total, exclude CANCELLED)
- Return fields: yearMonth, revenue, sorted desc.

Q2. Top 5 categories by units sold
- Unwind items, sum qty per items.category, return top 5.

Q3. Average order value (AOV) by loyalty tier
- Group by customer.loyalty_tier; average charges.grand_total for statuses in [PAID, PACKED, SHIPPED, DELIVERED, RETURNED].

Q4. Return rate by city
- % of orders with status == "RETURNED" per shipping.address.city. Show city, orders, returns, return_rate%.

Q5. Average delivery time (days) by city (DELIVERED only)

- Use order_date and shipping.delivered_at. Exclude nulls/negatives.

Q6. Customers with repeat purchases (>=3 orders)
- List customer.customer_id, name, total_orders, total_spent (sum of charges.grand_total excluding CANCELLED).

Q7. Orders that used coupons but paid COD
- Count and sample examples where charges.coupon != null and payment.method == "COD".

Q8. High value orders (top 10 by grand_total)
- Return _id, order_id, customer.name, charges.grand_total, shipping.address.city sorted desc.

Q9. Basket mix: category share within orders
- For each order, compute category breakdown (based on item line totals after item-level discounts). Return sample 10.

Q10. Geo query: orders within ~5km of Mumbai center (19.0760, 72.8777)
- Use $geoNear on shipping.geo (requires 2dsphere index). Return count + sample docs.

Q11. Discount analytics
- % of orders with any item-level discount (items.discount != null)
- Avg effective item-level discount across discounted lines.

Q12. Coupon effectiveness by tier
- For each customer.loyalty_tier, compute coupon usage rate and avg charges.coupon_discount.

Q13. Payment funnel from events
- Count how many orders reached each stage (ORDER_CREATED, PAYMENT_INITIATED, etc.) in order.

Q14. Review sentiment by category
- Using review.rating (1–5). Unwind items; average rating per items.category (ignore orders missing review).

Q15. Abandoned/failed payments
- status == "PLACED" AND last payment.transactions.status != "SUCCESS". Return order_id, customer, last transaction.