MongoDB – 100+ Lab Exercises Roadmap (Basic, Intermediate, Advanced)

☐ Basic Level (30+ Exercises)

Objective: Build foundational skills in NoSQL concepts, MongoDB basics, CRUD operations, and document structure.

☐ Introduction to MongoDB & NoSQL

- Install MongoDB on Windows/Linux/Mac.
- Understand NoSQL vs. RDBMS.
- Learn about documents, collections, databases.
- Use MongoDB shell (`mongosh`) and Compass UI.
- List databases and collections using `show dbs`, `show collections`.

☐ CRUD Operations

- Insert single and multiple documents (`insertOne`, `insertMany`).
- Retrieve data with `find()`, use projections.
- Update documents (`updateOne`, `updateMany`, `$set`, `$inc`).
- Delete documents with `deleteOne`, `deleteMany`.
- Use `replaceOne` and upserts.

☐ Querying Documents

- Use comparison operators (`$eq`, `$gt`, `$lt`, `$in`).
- Apply logical operators (`$and`, `$or`, `$not`).
- Filter documents using embedded document fields.
- Sort, skip, and limit results.
- Count documents in a collection.

☐ Data Modeling Basics

- Understand embedding vs. referencing.
- Design basic schemas using nested documents.

- Use `_id` fields and custom primary keys.
- Create collections with validation rules.
- Normalize vs. denormalize data structure in MongoDB.

☐ Intermediate Level (40+ Exercises)

Objective: Enhance capabilities in indexing, aggregation, data modeling, schema design, and application integration.

☐ Indexing & Performance

- Create and drop indexes.
- Use `explain()` to analyze query plans.
- Create compound indexes and unique indexes.
- Create text indexes for full-text search.
- Optimize read performance with indexes.

☐ Aggregation Framework

- Use `$match`, `$group`, `$sort`, `$project`.
- Work with `$lookup` for joins between collections.
- Build pipelines with multiple stages.
- Use `$unwind` for array flattening.
- Perform conditional operations with `$cond`, `$ifNull`.

☐ Schema Design & Validation

- Apply schema validation using `$jsonSchema`.
- Set up required fields, data types, and enums.
- Apply data integrity rules at collection level.
- Implement versioning for documents.
- Design schema for multi-tenant applications.

☐ Application Integration

- Connect MongoDB with Node.js (using Mongoose).
- Create RESTful API with Express + MongoDB.
- Integrate MongoDB with Python using PyMongo.

- Store user session data in MongoDB.
- Use MongoDB as a backend for MERN/MEVN stack apps.

☐ Advanced Level (40+ Exercises)

Objective: Master MongoDB for production use—covering sharding, replication, security, backup strategies, and advanced analytics.

☐ Replication & High Availability

- Set up a replica set.
- Configure primary-secondary nodes.
- Test automatic failover.
- Add and remove nodes from replica set.
- Use `rs.status()` and `rs.initiate()`.

☐ Sharding & Horizontal Scaling

- Configure a sharded cluster.
- Understand shard key selection strategy.
- Split and move chunks manually.
- Use `balancer` and monitor data distribution.
- Analyze sharded performance and consistency.

☐ Security & Access Control

- Set up authentication (`SCRAM`, `x.509`).
- Create roles and assign privileges.
- Use IP whitelisting and firewalls.
- Enable and configure TLS/SSL encryption.
- Audit user activity and queries.

☐ Backup & Disaster Recovery

- Use `mongodump` and `mongorestore` for snapshots.
- Backup and restore replica sets.
- Configure cloud backups using Atlas.
- Test disaster recovery with restore scenarios.

- Automate backups with scripts and cron.

☐ Monitoring & DevOps

- Monitor metrics using `mongostat`, `mongotop`.
- Use MongoDB Atlas monitoring dashboard.
- Integrate with Prometheus + Grafana.
- Set up logging and log rotation.
- Automate provisioning with Docker, Kubernetes.

☐ Capstone Projects

- Design and build a NoSQL database for a real-time chat app.
- Create an e-commerce backend with advanced filtering and aggregation.
- Deploy a sharded cluster on the cloud (AWS/GCP/Azure).
- Secure and monitor a production-grade MongoDB instance.
- Integrate MongoDB with Kafka or RabbitMQ for event-driven architecture.

☐ Tools & Technologies

- **MongoDB Shell (`mongosh`)**, Compass UI
- **Mongoose (Node.js ODM)**, PyMongo
- **MongoDB Atlas**, Robo 3T, Studio 3T
- **Docker, Kubernetes, Helm**
- **Prometheus, Grafana, Elastic Stack (ELK)**