# NoSQL Lab (MongoDB) - 2.5 Hours

## 1. Introduction to NoSQL Databases (20 mins)

Why NoSQL? Scalability, high availability, flexibility, and performance.  
NoSQL Database Types:  
- Key-Value Store (e.g., Redis, DynamoDB)  
- Document Store (e.g., MongoDB, CouchDB)  
- Column-Family Store (e.g., Cassandra, HBase)  
- Graph Database (e.g., Neo4j, ArangoDB)  
MongoDB Overview:  
- A document-oriented NoSQL database.  
- Uses BSON (Binary JSON) format.  
- Schema-less (flexible data models).  
- Supports horizontal scaling & replication.

## 2. Installing MongoDB (20 mins)

Windows Installation:  
1. Download MongoDB from https://www.mongodb.com/try/download/community  
2. Install MongoDB Community Edition.  
3. Add MongoDB to the system path.  
4. Start MongoDB service using `mongod`.  
5. Open another terminal and connect using `mongo`.  
  
Linux Installation:  
```sh  
sudo apt update  
sudo apt install -y mongodb  
sudo systemctl start mongodb  
sudo systemctl enable mongodb  
mongo  
```

## 3. Basic CRUD Operations (40 mins)

Task 1: Creating a Database & Collection  
```sh  
use StudentDB  
db.students.insertOne({ name: 'Ali Khan', age: 20, course: 'BSCS', semester: 4 })  
```

Task 2: Retrieving Data  
```sh  
db.students.find().pretty()  
```

Task 3: Updating Data  
```sh  
db.students.updateOne({ name: 'Ali Khan' }, { $set: { age: 21 } })  
```

Task 4: Deleting Data  
```sh  
db.students.deleteOne({ name: 'Sara Ahmed' })  
```

## 4. Advanced MongoDB Concepts (45 mins)

Task 5: Using Aggregation in MongoDB  
```sh  
db.students.aggregate([ { $group: { \_id: '$semester', avgMarks: { $avg: '$marks' } } } ])  
```

Task 6: Creating an Index  
```sh  
db.students.createIndex({ name: 1 })  
```

## 5. Connecting MongoDB with Python (PyMongo) (30 mins)

Install PyMongo:  
```sh  
pip install pymongo  
```  
  
Python Code to Insert and Retrieve Data:  
```python  
import pymongo  
client = pymongo.MongoClient('mongodb://localhost:27017/')  
db = client['StudentDB']  
collection = db['students']  
collection.insert\_one({'name': 'Ali', 'age': 21, 'semester': 4})  
students = collection.find()  
for student in students:  
 print(student)  
```

## 6. Assignment for Students

1. Install MongoDB and MongoDB Compass.  
2. Create a database for a Library Management System.  
3. Insert at least 10 books (title, author, year, category).  
4. Implement CRUD operations using MongoDB shell and PyMongo.  
5. Apply aggregation to find the average publication year.  
6. Use indexing to speed up queries.  
7. Export the database using:  
```sh  
mongodump --db LibraryDB --out /path/to/backup  
```

## 7. Lab Submission & Evaluation Criteria

- Correct installation (10%)  
- Successful CRUD operations (20%)  
- Aggregation & Indexing (20%)  
- Python connectivity (30%)  
- Report submission with screenshots (20%)