**Report: Student Database Management with MongoDB (CRUD Operations)**

**1. Dataset Design**

I created **a database named StudentDB** and a **collection named Students**. The dataset contains **15 student records**, each with the following fields:

**studentID:** Unique identifier for each student.

**name:** Full name of the student.

**course:** The course enrolled (e.g., Computer Science, Mathematics, Physics, Chemistry, Biology).

**marks:** Marks obtained by the student.

**city:** City of residence.

**email:** Contact email of the student.

================================================================================

**I ensured:**

Variety of courses (5 different ones).

A mix of marks (below 40, around average, and above 90) so that queries like delete below 40 and top 3 students show meaningful results.

Students distributed across different cities to demonstrate city-based queries.

==============================================================================

**2. Queries Executed**

**Basic Queries**

Display all student records.

Retrieve students in a specific course (Computer Science).

Find students with marks > 75.

Retrieve students from a specific city (Delhi).

**Update & Delete**

Updated one student’s email.

Increased marks by 10 for all students in Computer Science.

Deleted students with marks < 40.

**Aggregation & Analysis**

Counted the number of students per course.

Found the average marks per course (rounded to one decimal).

Retrieved top 3 students with the highest marks.

Sorted all students by marks in descending order.

**3. Insights**

**Course Distribution:** Computer Science had the highest number of students, showing its popularity.

**Top Performers:** Sneha Reddy (Computer Science), Priya Verma (Mathematics), and Vikram Singh (Chemistry) emerged as top students.

**Weak Performers:** Isha Gupta (Chemistry) scored below 40 and was deleted in cleanup.

**Average Marks:** Physics had moderate performance, while Mathematics showed the highest averages, making it the best-performing course overall.

**4. Export & Reporting**

The dataset was exported to students.json excluding MongoDB’s default \_id field for cleaner representation.

All steps and outputs have been demonstrated in Google Colab, connected to MongoDB Atlas.

**Conclusion**

This project provided hands-on practice with MongoDB operations such as insert, find, update, delete, and aggregation. Exporting the dataset and analyzing performance across courses showed how MongoDB can effectively manage and query structured educational data.