// 1. Create Collections and Insert Sample Data

// Switch to the 'school' database

use school

// Insert sample data into the 'Teachers' collection

db.Teachers.insertMany([

{ Tname: "John", dno: 1, dname: "Computer", experience: 5, salary: 12000, date\_of\_joining: new Date("2018-01-01") },

{ Tname: "Doe", dno: 2, dname: "IT", experience: 7, salary: 11000, date\_of\_joining: new Date("2017-01-01") },

{ Tname: "Jane", dno: 3, dname: "E&TC", experience: 8, salary: 9000, date\_of\_joining: new Date("2016-01-01") },

{ Tname: "Smith", dno: 4, dname: "Computer", experience: 3, salary: 8000, date\_of\_joining: new Date("2020-01-01") },

{ Tname: "Praveen", dno: 5, dname: "IT", experience: 9, salary: 13000, date\_of\_joining: new Date("2015-01-01") }

]);

// Insert sample data into the 'Students' collection

db.Students.insertMany([

{ Sname: "Alice", roll\_no: 1, class: "10th" },

{ Sname: "Bob", roll\_no: 2, class: "9th" },

{ Sname: "xyz", roll\_no: 3, class: "8th" }

]);

// 2. Find the information about all teachers

db.Teachers.find().pretty();

// 3. Find the information about all teachers of the Computer department

db.Teachers.find({ dname: "Computer" }).pretty();

// 4. Find the information about all teachers of Computer, IT, and E&TC departments

db.Teachers.find({ dname: { $in: ["Computer", "IT", "E&TC"] } }).pretty();

// 5. Find the information about all teachers of Computer, IT, and E&TC departments with salary >= 10000

db.Teachers.find({ dname: { $in: ["Computer", "IT", "E&TC"] }, salary: { $gte: 10000 } }).pretty();

// 6. Find student information where roll\_no = 2 or Sname = "xyz"

db.Students.find({ $or: [{ roll\_no: 2 }, { Sname: "xyz" }] }).pretty();

// 7. Update the experience of teacher 'Praveen' to 10 years; if not found, insert as a new entry

db.Teachers.updateOne(

{ Tname: "Praveen" },

{ $set: { experience: 10 } },

{ upsert: true }

);

// 8. Update the department name of all teachers in IT department to COMP

db.Teachers.updateMany(

{ dname: "IT" },

{ $set: { dname: "COMP" } }

);

// 9. Find the teacher's name and their experience from the Teachers collection

db.Teachers.find({}, { Tname: 1, experience: 1, \_id: 0 }).pretty();

// 10. Delete all documents from the Teachers collection where the department is IT

db.Teachers.deleteMany({ dname: "IT" });

// 11. Display the first 3 documents in the Teachers collection in ascending order by Tname

db.Teachers.find().sort({ Tname: 1 }).limit(3).pretty();