// ------------------------------

// Indexing Assignment - 1: students collection

// ------------------------------

// Step 1: Create Database and Insert Dummy Data

use myDatabase

db.createCollection("students");

db.students.insertMany([

{ name: "Raj", major: "Computer Science", gpa: 3.5, email: "raj@example.com" },

{ name: "Priya", major: "Mathematics", gpa: 3.8, email: "priya@example.com" },

{ name: "Anil", major: "Physics", gpa: 3.2, email: "anil@example.com" },

{ name: "Sita", major: "Chemistry", gpa: 3.9, email: "sita@example.com" }

]);

// Step 2: Create a single-field index on the name field

db.students.createIndex({ name: 1 });

// Verify index creation

db.students.getIndexes();

// Step 3: Create a compound index on the major and gpa fields

db.students.createIndex({ major: 1, gpa: 1 });

// Verify index creation

db.students.getIndexes();

// Step 4: Create a unique index on the email field

db.students.createIndex({ email: 1 }, { unique: true });

// Verify index creation

db.students.getIndexes();

// Step 5: Remove duplicate entries based on the email field

db.students.aggregate([

{

$group: {

\_id: "$email",

uniqueStudent: { $first: "$$ROOT" }

}

},

{

$replaceRoot: { newRoot: "$uniqueStudent" }

},

{

$out: "students"

}

]);

// Step 6: Drop the unique index on the email field

db.students.dropIndex({ email: 1 });

// Verify index removal

db.students.getIndexes();

// Step 7: Drop the compound index on the major and gpa fields

db.students.dropIndex({ major: 1, gpa: 1 });

// Verify index removal

db.students.getIndexes();

// Step 8: Retrieve and print the index statistics for the students collection

db.students.stats();

// ------------------------------

// Indexing Assignment - 2: students\_club collection

// ------------------------------

// Step 1: Create Database and Insert Dummy Data

use myDatabase

db.createCollection("students\_club");

db.students\_club.insertMany([

{ club\_name: "Robotics", member\_count: 50, email: "raj@example.com" },

{ club\_name: "Math Club", member\_count: 30, email: "priya@example.com" },

{ club\_name: "Physics Club", member\_count: 20, email: "anil@example.com" },

{ club\_name: "Chemistry Club", member\_count: 40, email: "sita@example.com" }

]);

// Step 2: Create a single-field index on the club\_name field

db.students\_club.createIndex({ club\_name: 1 });

// Verify index creation

db.students\_club.getIndexes();

// Step 3: Create a compound index on the club\_name and member\_count fields

db.students\_club.createIndex({ club\_name: 1, member\_count: 1 });

// Verify index creation

db.students\_club.getIndexes();

// Step 4: Create a unique index on the email field

db.students\_club.createIndex({ email: 1 }, { unique: true });

// Verify index creation

db.students\_club.getIndexes();

// Step 5: Remove duplicate entries based on the email field

db.students\_club.aggregate([

{

$group: {

\_id: "$email",

uniqueClubMember: { $first: "$$ROOT" }

}

},

{

$replaceRoot: { newRoot: "$uniqueClubMember" }

},

{

$out: "students\_club"

}

]);

// Step 6: Drop the unique index on the email field

db.students\_club.dropIndex({ email: 1 });

// Verify index removal

db.students\_club.getIndexes();

// Step 7: Drop the compound index on the club\_name and member\_count fields

db.students\_club.dropIndex({ club\_name: 1, member\_count: 1 });

// Verify index removal

db.students\_club.getIndexes();

// Step 8: Retrieve and print the index statistics for the students\_club collection

db.students\_club.stats();