const { MongoClient } = require('mongodb');

const carDrivers = [

    {

        name: "John Doe",

        vehicleType: "Sedan",

        isAvailable: true,

        rating: 4.9

    },

    {

        name: "Alice Smith",

        vehicleType: "SUV",

        isAvailable: false,

        rating: 4.4

    },

];

console.log(carDrivers);

const newDriver = {

    name: "Syafiq Iman",

    vehicleType: "Convertible",

    isAvailable: true,

    rating: 4.3

};

carDrivers.push(newDriver);

console.log(carDrivers);

async function main() {

    const uri = "mongodb://localhost:27017";

    const client = new MongoClient(uri);

    try {

        console.time("Connection Time");  // Start the timer

        await client.connect();

        console.log("Connected to MongoDB");

        console.timeEnd("Connection Time");  // End the timer

        const db = client.db("testDB");

        const carDriversCollection = db.collection("carDrivers");

        //CREATE: Insert each driver in the carDrivers array

        for (const driver of carDrivers) {

            const result = await carDriversCollection.insertOne(driver);

            console.log(`New driver inserted with result: ${result}`);

        }

        //READ: Find all drivers with rating >= 4.5 and isAvailable = true

        const availableDrivers = await db.collection("carDrivers").find({

            isAvailable: true,

            rating: {$gte: 4.5}

        }).toArray();

        console.log("available drivers:", availableDrivers);

        //UPDATE: Update the rating of a driver by incrementing them by 0.1 based on his/her name

       /\* const updateResult = await db.collection('carDrivers').updateOne(

            { name: "John Doe"},

            { $inc: { rating: 0.1}}

        );

        console.log(`Driver updated with result: ${updateResult}`)\*/

        //UPDATE: Update the rating of all drivers by incrementing them by 0.1 if they are available

        const updateResult = await db.collection('carDrivers').updateMany(

            { isAvailable: true},

            { $inc: { rating: 0.1}}

        );

        console.log(`Driver updated with result: ${updateResult}`)

        //DELETE: Delete all drivers by referring rating greater than or equal to 4.0

        let deleteResult = await db.collection('carDrivers').deleteMany({isAvailable: false});

        console.log(`Driver deleted with result: ${deleteResult}`);

        deleteResult = await db.collection('carDrivers').deleteMany(

            { rating: {$gte: 4.0}}

        );

        console.log(`Driver deleted with result: ${deleteResult}`)

    } catch (err) {

        console.error("Error:", err);

    } finally {

        await client.close();

    }

}

main();