using System;

using System.Collections.Generic;

using System.Collections.ObjectModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using MongoDB.Bson;

using MongoDB.Driver;

namespace Beadando

{

internal class Program

{

static void Main(string[] args)

{

// MongoDB adatbázis csatlakozás(a sáját jelszavamat <password>-el kicseréltem)

var client = new MongoClient("mongodb+srv://marciberecz02:<password>@mongo.uyzktqs.mongodb.net/");

var database = client.GetDatabase("Autok\_Beadando");

// Kollekciók hivatkozása

var carsCollection = database.GetCollection<BsonDocument>("Autok");

//1. Összes adat lekérdezése

var carsFilter = Builders<BsonDocument>.Filter.Empty;

var cars = carsCollection.Find(carsFilter).ToList();

Console.WriteLine("Összes adat:");

foreach (var car in cars)

{

Console.WriteLine(car);

}

//2.autó adat módosítás

var updateFilter = Builders<BsonDocument>.Filter.Eq("name", "Kiss Ádam");

var update = Builders<BsonDocument>.Update.Set("cars.0.miles", 35000);

carsCollection.UpdateOne(updateFilter, update);

var filter = Builders<BsonDocument>.Filter.Empty;

var people = carsCollection.Find(filter).ToList();

Console.WriteLine("\nMódosított adatok");

foreach (var person in people)

{

Console.WriteLine(person);

}

//3.adat törlés

var deleteFilter = Builders<BsonDocument>.Filter.Eq("name", "Kovács István");

carsCollection.DeleteMany(deleteFilter);

//4.új személy hozzáadás

Console.WriteLine("\núj személy hozzáadása");

var newPerson = new BsonDocument

{

{ "name", "Kovács István" },

{ "email", "kovacsistvan@gmail.com" },

{ "tel", "123(456)789-10-11" },

{ "age", 30 },

{ "cars", new BsonArray

{

new BsonDocument

{

{ "model", "Ford Mustang" },

{ "year", 2019 },

{ "color", "blue" },

{ "miles", 15000 }

}

}

}

};

carsCollection.InsertOne(newPerson);

//új személy kiírása

Console.WriteLine("Új személy:");

var newpersonFilter = Builders<BsonDocument>.Filter.Eq("name", "Kovács István");

var newperson = carsCollection.Find(newpersonFilter).ToList();

Console.WriteLine(newperson.ToJson());

//5.Legfitalabb és legidősebb személyek

var legtobb = carsCollection.Find(new BsonDocument()).Sort(Builders<BsonDocument>.Sort.Descending("age")).Limit(1).FirstOrDefault();

var legkevesebb = carsCollection.Find(new BsonDocument()).Sort(Builders<BsonDocument>.Sort.Ascending("age")).Limit(1).FirstOrDefault();

Console.WriteLine("Legidősebb személy:");

Console.WriteLine(legtobb.ToJson());

Console.WriteLine("Legfiatalabb személy:");

Console.WriteLine(legkevesebb.ToJson());

Console.ReadKey();

}

}

}