

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Лабораторна робота №14  
з дисципліни  
«ОБДЗ»

Виконав:  
Студент групи КН – 209  
Гречух Тарас

Перевірила:  
Мельникова Н.І.

Львів 2020

**Тема:** Розробка бази даних типу NoSQL

**Мета:** Здобуття практичних навичок створення та обробки бази даних типу NoSQL на прикладі СУБД MongoDB.

**Короткі теоретичні відомості:**

Функціональні можливості:

- узгодженість даних
- транзакції
- доступність
- можливості запитів
- масштабування

Типи значень:

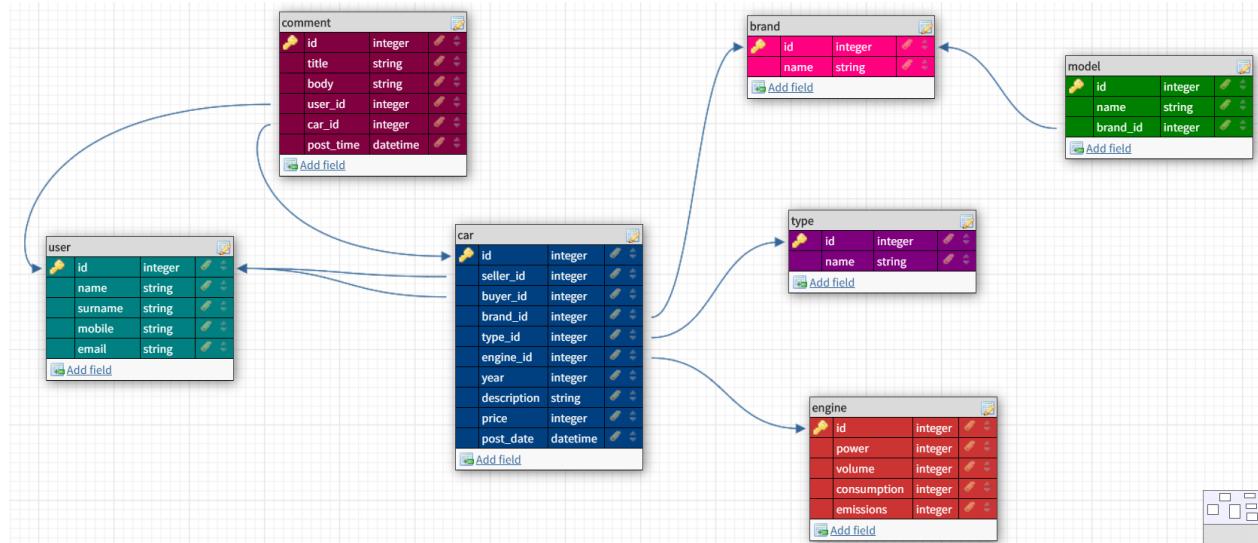
- String
- Array
- Binary data
- Boolean
- Date
- Double
- Integer
- JavaScript
- Min key/Max key
- Null
- Object
- ObjectId
- Regular expression
- Symbol
- Timestamp

**Завдання:**

1. Розробити схему бази даних на основі предметної області з лабораторної роботи №1 у спосіб, що застосовується в СУБД MongoDB..
2. Перетворити сутності діаграми БД, розробленої для лабораторної роботи №1, у структури, прийнятні для обробки в MongoDB.
3. Забезпечити реалізацію функцій редагування, додавання та вилучення інформації в «сущність».
4. Підготувати та захистити звіт до лабораторної роботи.

## Хід роботи:

Реляціональна база даних з першої лабораторної роботи:



Оскільки використовується база даних NoSql, нам варто згрупувати деякі таблиці, та якщо є проміжні – позбутися них. Отже залишаємо таблиці car, user, comment, brand. Усі решта таблиці згрупуємо, а саме type та engine – як додаткові поля таблиці автомобіля, model, як додаткове поле таблиці брендів.

1. Створення бази даних carsell та додавання в неї колекцій car, user, comment, brand:

```

> use carsell
switched to db carsell
> db.createCollection("user")
{ "ok" : 1 }
> db.createCollection("car")
{ "ok" : 1 }
> db.createCollection("comment")
{ "ok" : 1 }
> db.createCollection("brand")
{ "ok" : 1 }
  
```

Перевірити виконання наших дій можна в програмі MongoDB Compass:

The screenshot shows the MongoDB Compass interface with the 'Databases' tab selected. A green button labeled 'CREATE DATABASE' is visible. Below it is a table with four rows, each representing a database:

Database Name	Storage Size	Collections	Indexes
admin	20.0KB	0	1
carsell	32.0KB	4	4
config	24.0KB	0	2
local	36.0KB	1	1

Each row has a small trash can icon in the last column.

2. Забезпечення реалізації функцій редагування, додавання та вилучення інформації в «сущність».

Додаємо користувачів в колекцію user командою **db.user.insertOne()**.

```
> db.user.insertOne({ "_id":1, "name": "Taras", "surname": "Hrechukh", "mobile": "067-37-36-436", "email": "tarash281@gmail.com" })
{ "acknowledged" : true, "insertedId" : 1 }

> db.user.insertOne({ "_id":2, "name": "Narmas", "surname": "Topchan", "mobile": "007-77-77-007", "email": "just_top@gmail.com" })
{ "acknowledged" : true, "insertedId" : 2 }
> db.user.insertOne({ "_id":3, "name": "Star", "surname": "Lord", "mobile": "111-11-11-111", "email": "starlord@gmail.com" })
{ "acknowledged" : true, "insertedId" : 3 }
>
```

Командою **db.user.find()** переглядаємо наявних користувачів у колекції user:

```
> db.user.find()
{ "_id" : 1, "name" : "Taras", "surname" : "Hrechukh", "mobile" : "067-37-36-436", "email" : "tarash281@gmail.com" }
{ "_id" : 2, "name" : "Narmas", "surname" : "Topchan", "mobile" : "007-77-77-007", "email" : "just_top@gmail.com" }
{ "_id" : 3, "name" : "Star", "surname" : "Lord", "mobile" : "111-11-11-111", "email" : "starlord@gmail.com" }
>
```

Аналогічні операції проводимо з рештою колекцій:

Колекція car:

```
> db.car.insertOne({ "_id": 1, "seller_id": 1, "buyer_id": 2, "brand_id" : 1, "type": "sedan", "year": 2010, "price": 8000, "post_date": "2019-04-11" })
{ "acknowledged" : true, "insertedId" : 1 }
> db.car.insertOne({ "_id": 2, "seller_id": 2, "buyer_id": 3, "brand_id" : 2, "type": "coupe", "year": 2019, "price": 18000, "post_date": "2019-07-11" })
{ "acknowledged" : true, "insertedId" : 2 }
> db.car.insertOne({ "_id": 3, "seller_id": 3, "buyer_id": 1, "brand_id" : 3, "type": "coupe", "year": 2015, "price": 250000, "post_date": "2020-02-17" })
{ "acknowledged" : true, "insertedId" : 3 }
>
```

Колекція brand:

```
> db.brand.insertOne({_id: 1, "name": "BMW"})
{ "acknowledged" : true, "insertedId" : 1 }
> db.brand.insertOne({_id: 2, "name": "Mercedes"})
{ "acknowledged" : true, "insertedId" : 2 }
> db.brand.insertOne({_id: 3, "name": "Koenigsegg"})
{ "acknowledged" : true, "insertedId" : 3 }
> db.brand.insertOne({_id: 4, "name": "Toyota"})
{ "acknowledged" : true, "insertedId" : 4 }
> db.brand.insertOne({_id: 5, "name": "Volkswagen"})
{ "acknowledged" : true, "insertedId" : 5 }
>
```

Колекція comment:

```
{ db.comment.insertOne({_id: 1, "title": "My first comment", "body": "So good car and seller, want to buy it as soon as possible", "user_id": 1, "car_id": 2, "post_time": "2019-07-04"})
{ "acknowledged" : true, "insertedId" : 1 } My first comment", "body": "So good car and seller, want to buy it as soon as possible", "user_id": 1, "car_id": 2, "post_time": "2019-07-04"})
> db.comment.insertOne({_id: 2, "title": "Could be better", "body": "So good car and seller, but announcement is boring!", "user_id": 3, "car_id": 3, "post_time": "2019-05-23"})
{ "acknowledged" : true, "insertedId" : 2 }
> db.comment.insertOne({_id: 3, "title": "Nice car!", "body": "Nice state and price, thinking about buying it...", "user_id": 2, "car_id": 3, "post_time": "2019-12-02"})
{ "acknowledged" : true, "insertedId" : 3 }
> -
```

## Редагування інформації:

Змінимо рік випуску авто під номером 2 в колекції car з 2019 на 1998:

```
> db.car.updateOne({_id: 2}, {$set: {"year": 1998}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.car.find({_id: 2})
{ "_id" : 2, "seller_id" : 2, "buyer_id" : 3, "brand_id" : 2, "type" : "coupe", "year" : 1998, "price" : 18000, "post_date" : "2019-07-11" }
>
```

Змінимо електронну пошту користувача під номером 3:

```
> db.user.updateOne({_id: 3}, {$set: {"email": "updatedmail@gmail.com"})}
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.user.find({_id: 3})
{ "_id" : 3, "name" : "Star", "surname" : "Lord", "mobile" : "111-11-11-111", "email" : "updatedmail@gmail.com" }
> -
```

## Видалення інформації:

Видалимо бренд Volkswagen з колекції brand:

```
> db.brand.deleteOne({"name": "Volkswagen"})
{ "acknowledged" : true, "deletedCount" : 1 }
> db.brand.find()
{ "_id" : 1, "name" : "BMW" }
{ "_id" : 2, "name" : "Mercedes" }
{ "_id" : 3, "name" : "Koenigsegg" }
{ "_id" : 4, "name" : "Toyota" }
>
```

Видалимо усі коментарі які писалися до оголошення 3:

```
> db.comment.deleteMany({ "car_id": 3 })
{ "acknowledged" : true, "deletedCount" : 2 }
> db.comment.find()
{ "_id" : 1, "title" : "My first comment", "body" : "So good car and seller, want to buy it as soon as possible", "user_id" : 1, "car_id" : 2, "post_time" : "2019-07-04" }
>
```

Додамо декілька нових брендів та видалимо усі які починаються на літеру “Т”:

```
> db.brand.insertOne({ "_id": 5, "name": "Tesla" })
{ "acknowledged" : true, "insertedId" : 5 }
> db.brand.insertOne({ "_id": 6, "name": "Subaru" })
{ "acknowledged" : true, "insertedId" : 6 }
> db.brand.insertOne({ "_id": 7, "name": "TinyCar" })
{ "acknowledged" : true, "insertedId" : 7 }
> db.brand.insertOne({ "_id": 8, "name": "Teranosaurus" })
{ "acknowledged" : true, "insertedId" : 8 }
> db.brand.insertOne({ "_id": 9, "name": "Fiat" })
{ "acknowledged" : true, "insertedId" : 9 }
> db.brand.find()
{ "_id" : 1, "name" : "BMW" }
{ "_id" : 2, "name" : "Mercedes" }
{ "_id" : 3, "name" : "Koenigsegg" }
{ "_id" : 4, "name" : "Toyota" }
{ "_id" : 5, "name" : "Tesla" }
{ "_id" : 6, "name" : "Subaru" }
{ "_id" : 7, "name" : "TinyCar" }
{ "_id" : 8, "name" : "Teranosaurus" }
{ "_id" : 9, "name" : "Fiat" }
>
```

```
> db.brand.remove({ "name" : /T\w+/i })
WriteResult({ "nRemoved" : 4 })
> db.brand.find()
{ "_id" : 1, "name" : "BMW" }
{ "_id" : 2, "name" : "Mercedes" }
{ "_id" : 3, "name" : "Koenigsegg" }
{ "_id" : 6, "name" : "Subaru" }
{ "_id" : 9, "name" : "Fiat" }
>
```

Як згадувалося вище усі зміни можна переглянути в MongoDB Compass:

_id	name
1	BMW
2	Mercedes
3	Koenigsegg
6	Subaru
9	Fiat

**Висновок:** На цій лабораторній роботі я здобув практичних навичок створення та обробки бази даних типу NoSQL на прикладі СУБД MongoDB.