

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота №14

з дисципліни
«Організація баз даних та знань»

Виконав:

студент групи КН-208

Гірник Ю.В.

Викладач:

к.т.н.

Мельникова Н.І.

Львів – 2020 р.

Лабораторна робота №14

Розробка бази даних типу NoSQL

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

Хід роботи

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 «сутність».
4. Підготувати та захистити звіт до лабораторної роботи.

1. Розробити схему бази даних на основі предметної області з лабораторної роботи №1 у спосіб, що застосовується в СУБД MongoDB.



Рис.1. Модель БД з лабораторної роботи №1

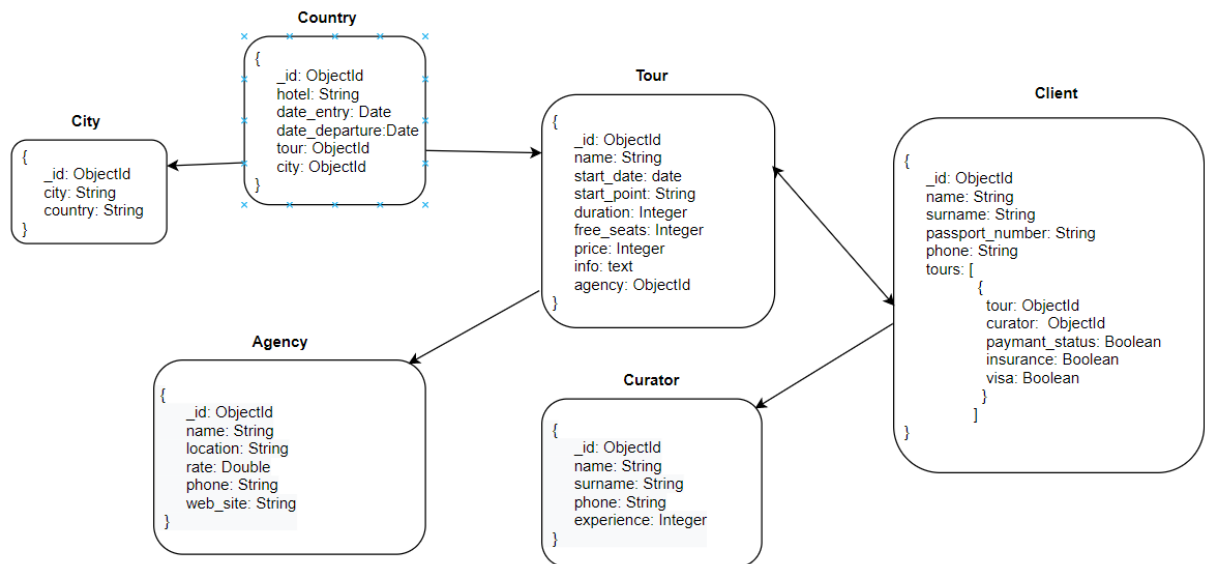


Рис.2. Модель БД NoSQL

2. Перетворити сутності діаграми БД, розробленої для лабораторної роботи №1, у структури, прийнятні для обробки в MongoDB.

```

> use travel_agency
switched to db travel_agency
> db.createCollection('city')
{ "ok" : 1 }
> db.createCollection('country')
{ "ok" : 1 }
> db.createCollection('tour')
{ "ok" : 1 }
> db.createCollection('agency')
{ "ok" : 1 }
> db.createCollection('curator')
{ "ok" : 1 }
> db.createCollection('client')
{ "ok" : 1 }

```

3. Забезпечити реалізацію функцій редагування, додавання та вилучення інформації в «сутність».

Заповнимо наші колекції даними:

1) “Agency”

```

> db.agency.insertMany([
... { '_id': 'agency1', 'name': 'TravelEveryDay', 'location': 'Lviv', 'rate': 3.4, 'phone': '+38096779987', 'web_site': 'travel.com' },
... { '_id': 'agency2', 'name': 'SkoleTravel', 'location': 'Skole', 'rate': 4.9, 'phone': '+38096123412', 'web_site': 'skoleT.com' },
... { '_id': 'agency3', 'name': 'IndexLife', 'location': 'Uzhhorod', 'rate': 2.4, 'phone': '+38092131244', 'web_site': 'Kokos.com' }])

```

2) “City”

```
> db.city.insertMany([
... {'_id':'city1', 'city': 'Відень', 'country':'Австрія'},
... {'_id':'city2', 'city': 'Львів', 'country':'Україна'},
... {'_id':'city3', 'city': 'Кошице', 'country':'Словаччина'}])_
```

3) “Curator”

```
> db.curator.insertMany([
... {'_id':'curator1', 'name': 'Ivan', 'surname': 'Popov', 'phone': '38091234123', 'experience': '3'},
... {'_id':'curator2', 'name': 'Mike', 'surname': 'Kok', 'phone': '38091212313', 'experience': '9'},
... {'_id':'curator3', 'name': 'Kokich', 'surname': 'Zhukich', 'phone': '38091234123', 'experience': '0'}])
```

4) “Tour”

```
> db.tour.insertMany([
... {'_id':'tour1', 'name':'Подорож до Відню', 'start_date': new Date('2020-08-12'), 'start_point': 'Lviv', 'duration': 4, 'free_seats': 30, 'price': 300, 'info': 'Неймовірний Відень', 'agency': 'agency1'},
... {'_id':'tour2', 'name':'Казковий Львів', 'start_date': new Date('2020-08-09'), 'start_point': 'Skole', 'duration': 2, 'free_seats': 40, 'price': 100, 'info': 'Дивовижні пам'ятки', 'agency': 'agency2'},
... {'_id':'tour3', 'name':'Унікальна Кошице', 'start_date': new Date('2020-07-02'), 'start_point': 'Lviv', 'duration': 3, 'free_seats': 15, 'price': 150, 'info': 'Словачка перлина', 'agency': 'agency2'},
... {'_id':'tour4', 'name':'Відчувай себе королем', 'start_date': new Date('2020-09-01'), 'start_point': 'Lviv', 'duration': 5, 'free_seats': 20, 'price': 400, 'info': 'Австрійські замки тебе чекають', 'agency': 'agency3'},
... {'_id':'tour5', 'name':'Львівська романтика', 'start_date': new Date('2020-10-02'), 'start_point': 'Uzhhorod', 'duration': 2, 'free_seats': 90, 'price': 120, 'info': 'Вищий ароматної кави', 'agency': 'agency3'}])_
```

5) “Country”

```
> db.country.insertMany([
... {'_id':'country1', 'hotel':'Victoria', 'date_entry': new Date('2020-08-12'), 'date_departure': new Date('2020-08-14'), 'tour': 'tour1', 'city':'city1'},
... {'_id':'country2', 'hotel':'SlavikoKoks', 'date_entry': new Date('2020-07-02'), 'date_departure': new Date('2020-07-03'), 'tour': 'tour3', 'city':'city3'},
... {'_id':'country3', 'hotel':'VienHotel', 'date_entry': new Date('2020-09-01'), 'date_departure': new Date('2020-09-06'), 'tour': 'tour4', 'city':'city1'},
... {'_id':'country4', 'hotel':'Fashion', 'date_entry': new Date('2020-10-02'), 'date_departure': new Date('2020-10-03'), 'tour': 'tour5', 'city':'city2'},
... {'_id':'country5', 'hotel':'Fashion', 'date_entry': new Date('2020-10-02'), 'date_departure': new Date('2020-10-03'), 'tour': 'tour5', 'city':'city2'}])
```

6) “Client” – тут на відміну від лабораторної роботи №1, в клієнті є поле tours, яке є масивом об'єктів.

```
> db.client.insertMany([
... {'_id':'client1', 'name': 'Ivan', 'surname': 'Ivanov', 'phone': '380912341123', 'passport_number': '2314214123124', 'tours:[
...   {'tour':'tour1', 'curator':'curator1', 'payment_status': true, 'insurance': true, 'visa': false},
...   {'tour':'tour2', 'curator':'curator2', 'payment_status': true, 'insurance': true, 'visa': false}]},
... {'_id':'client2', 'name': 'Pavlo', 'surname': 'Zorro', 'phone': '38091223123', 'passport_number': '23142141231242', 'tours:[
...   {'tour':'tour3', 'curator':'curator3', 'payment_status': true, 'insurance': false, 'visa': false}],},
... {'_id':'client3', 'name': 'Igor', 'surname': 'Chai', 'phone': '38091234123', 'passport_number': '23142141231223', 'tours:[
...   {'tour':'tour4', 'curator':'curator3', 'payment_status': false, 'insurance': true, 'visa': true}],},
... {'_id':'client4', 'name': 'Mike', 'surname': 'Kek', 'phone': '38045134123', 'passport_number': '23142141212324', 'tours:[
...   {'tour':'tour5', 'curator':'curator2', 'payment_status': true, 'insurance': false, 'visa': true}],},
... {'_id':'client5', 'name': 'Les', 'surname': 'Ivazhuk', 'phone': '38094534123', 'passport_number': '23142141212344', 'tours:[
...   {'tour':'tour2', 'curator':'curator1', 'payment_status': false, 'insurance': false, 'visa': false}]}])_
```

Проведемо вибірку даних:

1) Візьмемо одного клієнта

```
> client = db.client.findOne()
{
  "_id" : "client1",
  "name" : "Ivan",
  "surname" : "Ivanov",
  "phone" : "380912341123",
  "passport_number" : "2314214123124",
  "tours" : [
    {
      "tour" : "tour1",
      "curator" : "curator1",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    },
    {
      "tour" : "tour2",
      "curator" : "curator2",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    }
  ]
}
```

2) Тепер виберемо інформацію про тури, ObjectId яких зберігались в клієнті:

```
> db.tour.find({$or : [{ '_id': client.tours[0].tour }, { '_id': client.tours[1].tour } ]}).pretty()
{
  "_id" : "tour1",
  "name" : "Подорож до Відню",
  "start_date" : ISODate("2020-08-12T00:00:00Z"),
  "start_point" : "Lviv",
  "duration" : 4,
  "free_seats" : 30,
  "price" : 300,
  "info" : "Неймовірний Відень",
  "agency" : "agency1"
}
{
  "_id" : "tour2",
  "name" : "Казковий Львів",
  "start_date" : ISODate("2020-08-09T00:00:00Z"),
  "start_point" : "Skole",
  "duration" : 2,
  "free_seats" : 40,
  "price" : 100,
  "info" : "Дивовіжні пам'ятки",
  "agency" : "agency2"
}
```

3) Виберемо клієнтів, які оформили тур з _id "tour2"

```
> db.client.find({'tours.tour':'tour2'}).pretty()
{
  "_id" : "client1",
  "name" : "Ivan",
  "surname" : "Ivanov",
  "phone" : "380912341123",
  "passport_number" : "2314214123124",
  "tours" : [
    {
      "tour" : "tour1",
      "curator" : "curator1",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    },
    {
      "tour" : "tour2",
      "curator" : "curator2",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    }
  ]
}
{
  "_id" : "client5",
  "name" : "Les",
  "surname" : "Ivazhuk",
  "phone" : "38094534123",
  "passport_number" : "23142141212344",
  "tours" : [
    {
      "tour" : "tour2",
      "curator" : "curator1",
      "paymant_status" : false,
      "insurance" : false,
      "visa" : false
    }
  ]
}
```

Протестуємо оновлення даних:

1) Повністю модифікуємо тур з _id='tour3'

```
> db.agency.update({name : "IndexLife"}, {'_id':'agency3', 'name': 'NewName', 'location': 'Kosovo', 'rate': 1.4, 'phone': '+38092131244', 'web_site': 'Kokos.com'}, {upsert: true})
```

2) Перевіримо зміни:

```
> db.agency.find().pretty()
{
  "_id" : "agency1",
  "name" : "TravelEveryDay",
  "location" : "Lviv",
  "rate" : 3.4,
  "phone" : "+38096779987",
  "web_site" : "travel.com"
}
{
  "_id" : "agency2",
  "name" : "SkoleTravel",
  "location" : "Skole",
  "rate" : 4.9,
  "phone" : "+38096123412",
  "web_site" : "skoleT.com"
}
{
  "_id" : "agency3",
  "name" : "NewName",
  "location" : "Kosovo",
  "rate" : 1.4,
  "phone" : "+38092131244",
  "web_site" : "Kokos.com"
}
```

3) Тепер змінимо окрему поле:

```
> db.agency.update({name : "NewName"}, {$set: {rate : 3.0}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.agency.find({name: 'NewName'}).pretty()
{
  "_id" : "agency3",
  "name" : "NewName",
  "location" : "Kosovo",
  "rate" : 3,
  "phone" : "+38092131244",
  "web_site" : "Kokos.com"
}
```

4) Додамо тур до одного з клієнтів в масив та перевіримо:

```
> db.client.updateOne({'_id' : "client2"}, {$push: {tours: {'tour': 'tour2', 'curator': 'curator2', 'paymant_status': true, 'insurance': false, 'visa': true}}})
```

```

> db.client.find({'_id': 'client2'}).pretty()
{
  "_id" : "client2",
  "name" : "Pavlo",
  "surname" : "Zorro",
  "phone" : "38091223123",
  "passport_number" : "23142141231242",
  "tours" : [
    {
      "tour" : "tour3",
      "curator" : "curator3",
      "paymant_status" : true,
      "insurance" : false,
      "visa" : false
    },
    {
      "tour" : "tour2",
      "curator" : "curator2",
      "paymant_status" : true,
      "insurance" : false,
      "visa" : true
    }
  ]
}

```

Проведемо видалення даних:

- 1) Заберемо тур, який ми додали клієнту в минулому пункті:

```

> db.client.update({'_id' : "client2"}, {$pop: {tours: -1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.client.find({'_id': 'client2'}).pretty()
{
  "_id" : "client2",
  "name" : "Pavlo",
  "surname" : "Zorro",
  "phone" : "38091223123",
  "passport_number" : "23142141231242",
  "tours" : [
    {
      "tour" : "tour2",
      "curator" : "curator2",
      "paymant_status" : true,
      "insurance" : false,
      "visa" : true
    }
  ]
}

```

- 2) Видалимо другого клієнта та перевіримо:

```

> db.client.remove({'_id': 'client2'})
WriteResult({ "nRemoved" : 1 })

```



```
> db.client.find().pretty()
{
  "_id" : "client1",
  "name" : "Ivan",
  "surname" : "Ivanov",
  "phone" : "380912341123",
  "passport_number" : "2314214123124",
  "tours" : [
    {
      "tour" : "tour1",
      "curator" : "curator1",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    },
    {
      "tour" : "tour2",
      "curator" : "curator2",
      "paymant_status" : true,
      "insurance" : true,
      "visa" : false
    }
  ]
}
{
  "_id" : "client3",
  "name" : "Igor",
  "surname" : "Chai",
  "phone" : "38091234123",
  "passport_number" : "23142141231223",
  "tours" : [
    {
      "tour" : "tour4",
      "curator" : "curator3",
      "paymant_status" : false,
      "insurance" : true,
      "visa" : true
    }
  ]
}
```

Висновки: на цій лабораторній роботі я отримав практичні навички з роботою з NoSQL базами даних на прикладі MongoDB. Провів операції створення колекцій, вставки даних, оновлення та видалення