

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту



Лабораторна робота №14

з дисципліни “Організація баз даних та знань”

Виконав:

Романишин М.Р.

КН-211

Викладач:

Якимишин Х.М.

Лабораторна робота №14
з курсу “ОБДЗ”
на тему:
“Розробка бази даних типу NoSQL”

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

Хід роботи

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.

Виконання

1. Створення сутностей схеми з лабораторної роботи №1 в БД MongoDB

```
> use projmanagsys
switched to db projmanagsys
> db
projmanagsys
> db.createCollection('user')
{ "ok" : 1 }
> db.createCollection('task')
{ "ok" : 1 }
> db.createCollection('contacts')
{ "ok" : 1 }
> db.createCollection('project')
{ "ok" : 1 }
> db.createCollection('task_comment')
{ "ok" : 1 }
> db.createCollection('user_project_task')
{ "ok" : 1 }
```

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

Для цього використовуємо наступні команди:

`db.collectionName.insert({})` – для додавання 1 об'єкта в колекцію.

`db.collectionName.insert([{}])` – для додавання декількох об'єктів в колекцію.

```
> db.project.insert({projectId: 1, projectName: 'project1', customerCompany: 'qwerty'})
WriteResult({ "nInserted" : 1 })
> db.project.insert({projectId: 2, projectName: 'project2', customerCompany: 'Microsoft'})
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 1,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
> db.project.insert([projectId: 2, projectName: 'project2', customerCompany: 'Microsoft'], {projectId: 3, projectName: 'project3', customerCompany: 'Google'})
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
> db.contacts.insertMany([contacts_id: 1, email: '123@gmail.com', telegram: '1235'], {contacts_id: 2, email: 'kek@gmail.com', telegram: '5346'})
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("Sec7014bc406344f8ae76036"),
    ObjectId("Sec7014bc406344f8ae76037")
  ]
}
> db.task.insertMany([taskId: 1, deadlineTime: '2020-22-20', title: '123', description: 'desc1', status: 'in progress'], {taskId: 2, deadlineTime: '2020-06-01', title: 'title2', description: 'des2', status: 'in progress'})
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("Sec701dfc406344f8ae76038"),
    ObjectId("Sec701dfc406344f8ae76039")
  ]
}
> db.user.insertMany([userId: 1, contactsId: 1, surname: 'Rom', name: 'Myk', login: 'login1'], {userId: 2, contactsId: 2, surname: 'QME', name: 'RTV', login: 'login2'})
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("Sec7025cc406344f8ae7603a"),
    ObjectId("Sec7025cc406344f8ae7603b")
  ]
}
```

Активация Windows
Перейдите до розділу "Настройки", щоб активувати Windows.

```
> db.task_comment.insertMany([commentId: 1, userId: 1, taskId: 1, commentText: 'text1', postedTime: '2020-05-22'], {commentId: 2, userId: 2, taskId: 2, commentText: 'text2', postedTime: '2020-05-23'})
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("Sec702ebc406344f8ae7603c"),
    ObjectId("Sec702ebc406344f8ae7603d")
  ]
}
> db.user_project_task([userId: 1, taskId: 1, projectId: 1, userRole: 'admin'], {userId: 2, taskId: 2, projectId: 3, userRole: 'user'})
2020-05-22T01:40:01.298+0300 E QUERY [js] uncaught exception: TypeError: db.user_project_task is not a function :
js(shell):1:1
> db.user_project_task.insertMany([userId: 1, taskId: 1, projectId: 1, userRole: 'admin'], {userId: 2, taskId: 2, projectId: 3, userRole: 'user'})
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("Sec70357c406344f8ae7603e"),
    ObjectId("Sec70357c406344f8ae7603f")
  ]
}
```

За допомогою команди `db.collectionName.find().pretty()` отримуємо вміст колекції в форматovanому вигляді.

```

> db.project.find().pretty()
{
  "_id" : ObjectId("5ec6ffc1c406344f8ae76032"),
  "projectId" : 1,
  "projectName" : "project1",
  "customerCompany" : "qwerty"
}
{
  "_id" : ObjectId("5ec7001ac406344f8ae76033"),
  "projectId" : 2,
  "projectName" : "project2",
  "customerCompany" : "Microsoft"
}
{
  "_id" : ObjectId("5ec70056c406344f8ae76034"),
  "projectId" : 2,
  "projectName" : "project2",
  "customerCompany" : "Microsoft"
}
{
  "_id" : ObjectId("5ec70056c406344f8ae76035"),
  "projectId" : 3,
  "projectName" : "project3",
  "customerCompany" : "Google"
}

```

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

```

> db.user.update({login: 'login1'}, {$set: {login: 'mykola829'}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.user.find().pretty()
{
  "_id" : ObjectId("5ec7025cc406344f8ae7603a"),
  "userId" : 1,
  "contactsId" : 1,
  "surname" : "Rom",
  "name" : "Myk",
  "login" : "mykola829"
}
{
  "_id" : ObjectId("5ec7025cc406344f8ae7603b"),
  "userId" : 2,
  "contactsId" : 2,
  "surname" : "QWE",
  "name" : "RTY",
  "login" : "login2"
}
> db.user.remove({surname: 'QWE'})
WriteResult({ "nRemoved" : 1 })
> db.user.find().pretty()
{
  "_id" : ObjectId("5ec7025cc406344f8ae7603a"),
  "userId" : 1,
  "contactsId" : 1,
  "surname" : "Rom",
  "name" : "Myk",
  "login" : "mykola829"
}
> db.user.update({}, {$set: {adress: 'lol Str.'}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.user.find().pretty()
{
  "_id" : ObjectId("5ec7025cc406344f8ae7603a"),
  "userId" : 1,
  "contactsId" : 1,
  "surname" : "Rom",
  "name" : "Myk",
  "login" : "mykola829",
  "adress" : "lol Str."
}

```

Висновок: на цій лабораторній роботі було створено noSQL базу даних MongoDB на основі бази даних з лабораторної роботи №1. Було використано базові команди mongoDB, проведено давання, оновлення, видалення полів, їх відображення.