**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп’ютерних наук та кібернетики

**Звіт**

до інструкції по MongoDB

з дисципліни

**«**Проєктування та супровід баз даних та знань**»**

Виконали студенти 1-го курсу

Ходаков Максим

Сомов Максим

Київ – 2025

Посібник з MongoDB

MongoDB — це сучасна, потужна, документно-орієнтована база даних, створена для роботи з великими, неструктурованими або напівструктурованими даними. Вона належить до класу NoSQL систем і широко використовується в сучасних проєктах, які потребують гнучкості схеми, масштабованості та високої швидкодії.

# Вступ

У сучасному світі даних класичні реляційні бази даних, як-от MySQL або SQL Server, часто не можуть задовольнити потреби динамічних проєктів із непостійною структурою даних. MongoDB була створена як відповідь на ці виклики — її головна перевага полягає у зберіганні даних у форматі BSON (Binary JSON), який поєднує читабельність JSON і ефективність бінарного кодування.

MongoDB забезпечує:  
• Високу масштабованість (через шардінг);  
• Високу доступність (через реплікацію);  
• Гнучкість структури даних (без жорсткої схеми);  
• Потужний набір інструментів для аналітики та обробки інформації.

# 1. Основні концепції MongoDB

MongoDB не базується на таблицях і рядках, а використовує гнучку структуру документів. Ці документи мають формат JSON і можуть зберігати вкладені об’єкти, масиви, числа, рядки, дати тощо.

Основні елементи MongoDB:  
• \*\*Документ\*\* — це об’єкт JSON, що містить поля (ключі) та значення. Наприклад:  
```json  
{ name: "Іван", age: 25, city: "Київ" }  
```  
• \*\*Колекція\*\* — набір документів, що зберігаються разом (аналог таблиці в SQL).  
• \*\*База даних\*\* — контейнер для колекцій.  
• \*\*ObjectId\*\* — унікальний ідентифікатор документа.  
• \*\*BSON\*\* — внутрішній формат MongoDB, який розширює JSON додатковими типами (наприклад, date, binary, decimal).

# 2. Встановлення та запуск MongoDB

Існує два основні способи розгорнути MongoDB: локально або в хмарі через MongoDB Atlas.

### Встановлення локально:  
1. Перейдіть на сайт https://www.mongodb.com/try/download/community.  
2. Завантажте інсталятор для вашої операційної системи.  
3. Запустіть MongoDB Server (`mongod`) і клієнт (`mongosh`).  
4. Для графічного доступу використовуйте MongoDB Compass.

### Встановлення у хмарі (MongoDB Atlas):  
1. Зареєструйтеся на https://cloud.mongodb.com.  
2. Створіть кластер типу Free Tier (M0).  
3. Додайте користувача, IP-адресу та отримайте підключення.  
4. Використовуйте рядок підключення MongoDB SRV у своєму застосунку.

# 3. CRUD-операції

CRUD — це набір базових операцій у будь-якій базі даних: Create (створення), Read (читання), Update (оновлення), Delete (видалення).

### Створення:  
```js  
db.users.insertOne({ name: 'Олена', age: 28, city: 'Київ' });  
db.users.insertMany([{ name: 'Ігор' }, { name: 'Марія' }]);  
```  
### Читання:  
```js  
db.users.find({ city: 'Київ' });  
db.users.findOne({ name: 'Олена' });  
```  
### Оновлення:  
```js  
db.users.updateOne({ name: 'Олена' }, { $set: { age: 29 } });  
```  
### Видалення:  
```js  
db.users.deleteOne({ name: 'Марія' });  
```

# 4. Агрегаційний фреймворк

MongoDB має потужну систему агрегування, що дозволяє групувати, сортувати, фільтрувати та аналізувати дані.

```js  
db.sales.aggregate([  
 { $match: { status: 'complete' } },  
 { $group: { \_id: '$region', total: { $sum: '$amount' } } },  
 { $sort: { total: -1 } }  
]);  
```

Етапи агрегування (stages):  
• $match — фільтрує дані.  
• $group — групує за полем.  
• $sort — сортує.  
• $project — вибір та перейменування полів.  
• $lookup — реалізує JOIN між колекціями.  
• $limit, $skip — пагінація результатів.

# 5. Валідація структури документів

MongoDB дозволяє накладати обмеження на структуру документів за допомогою JSON Schema. Це допомагає контролювати типи полів та їх обов’язковість.

```js  
db.createCollection('products', {  
 validator: {  
 $jsonSchema: {  
 bsonType: 'object',  
 required: ['name', 'price'],  
 properties: {  
 name: { bsonType: 'string' },  
 price: { bsonType: 'double', minimum: 0 }  
 }  
 }  
 }  
});  
```

# 6. Реплікація і шардінг

MongoDB побудована з урахуванням горизонтального масштабування і безперебійності.

• \*\*Replica Set\*\* — це група серверів, які синхронізують дані. Один із них є primary, інші — secondary. У разі відмови primary система автоматично призначає новий.  
• \*\*Sharding\*\* — це розподіл даних між кількома вузлами для підвищення продуктивності.

# 7. Транзакції

Починаючи з MongoDB 4.0, система підтримує ACID-транзакції, подібно до SQL.

```js  
const session = db.getMongo().startSession();  
session.startTransaction();  
try {  
 session.getDatabase('bank').accounts.updateOne({ user: 'Іван' }, { $inc: { balance: -100 } });  
 session.getDatabase('bank').accounts.updateOne({ user: 'Марія' }, { $inc: { balance: 100 } });  
 session.commitTransaction();  
} catch (e) {  
 session.abortTransaction();  
}  
```

# 8. Практичний приклад: бібліотека книг

```js  
use library;  
db.books.insertMany([  
 { title: 'Кобзар', author: 'Шевченко', year: 1840 },  
 { title: 'Майстер і Маргарита', author: 'Булгаков', year: 1967 }  
]);  
db.books.find({ year: { $gt: 1900 } });  
```

# Глосарій MongoDB

BSON — двійкове представлення JSON-документів.  
Document — основна одиниця даних у MongoDB.  
Collection — набір документів, аналог таблиці.  
Replica Set — набір серверів MongoDB для реплікації.  
Sharding — розподіл даних між кількома вузлами.  
Pipeline — послідовність етапів агрегації.  
Index — структура даних для прискорення пошуку.

# Висновок

MongoDB — це база даних, яка поєднує в собі простоту, гнучкість і потужність. Вона підходить для стартапів, аналітичних систем, інтернет-магазинів і складних мікросервісів. Її головна перевага — збереження складних об’єктів у природному форматі JSON.