

Лабораторная работа: "Изучение методов хранения данных на основе NoSQL в MongoDB, Redis, Neo4j"

Цель работы:

Изучить и освоить методы хранения и работы с данными в NoSQL базах данных MongoDB, Redis и Neo4j. Научиться загружать данные из CSV файлов в указанные СУБД и выполнять базовые операции по работе с данными.

Оборудование и ПО:

- Операционная система Ubuntu
- Установленные пакеты для работы с NoSQL базами данных: MongoDB, Redis, Neo4j
- Язык программирования Python (с библиотеками pymongo, redis, neo4j)
- CSV файлы с данными

Теоретическая часть:

1. ****MongoDB:**** документо-ориентированная NoSQL база данных, где данные хранятся в формате JSON-подобных документов.
2. ****Redis:**** высокопроизводительная база данных типа "ключ-значение", часто используемая для кеширования и временного хранения данных.
3. ****Neo4j:**** графовая база данных, которая позволяет хранить данные в виде вершин и рёбер графа, что удобно для моделирования сложных взаимосвязей.

Ход работы:

Шаг 1: Установка и настройка MongoDB

1.1. Установите MongoDB:

```
```bash
sudo apt-get update
sudo apt-get install -y mongodb
```
```

1.2. Запустите MongoDB:

```
```bash
sudo systemctl start mongod
```
```

1.3. Загрузка данных из CSV в MongoDB:

```
```python
import csv
from pymongo import MongoClient

client = MongoClient('localhost', 27017)
```

```
db = client['lab_db']
collection = db['lab_collection']

with open('data.csv', mode='r') as file:
 reader = csv.DictReader(file)
 for row in reader:
 collection.insert_one(row)
...
```

## ##### Шаг 2: Установка и настройка Redis

### 2.1. Установите Redis:

```
``bash
sudo apt-get install redis-server
...
```

### 2.2. Запустите Redis:

```
``bash
sudo systemctl start redis
...
```

### 2.3. Загрузка данных из CSV в Redis:

```
``python
import csv
import redis

r = redis.Redis(host='localhost', port=6379, db=0)

with open('data.csv', mode='r') as file:
 reader = csv.reader(file)
 header = next(reader) # Считываем заголовок
 for row in reader:
 r.hmset(f"row:{row[0]}", {header[i]: row[i] for i in range(1, len(row))})
...
```

## ##### Шаг 3: Установка и настройка Neo4j

### 3.1. Установите Neo4j:

```
``bash
wget -O - https://debian.neo4j.com/neotechnology.gpg.key | sudo apt-key add -
echo 'deb https://debian.neo4j.com stable 4.x' | sudo tee
/etc/apt/sources.list.d/neo4j.list
sudo apt-get update
sudo apt-get install neo4j
...
```

### 3.2. Запустите Neo4j:

```
```bash
sudo systemctl start neo4j
```
```

### 3.3. Загрузка данных из CSV в Neo4j:

```
```python
from neo4j import GraphDatabase

driver = GraphDatabase.driver("bolt://localhost:7687", auth=("neo4j",
"password"))

def create_node(tx, label, properties):
    query = f"CREATE (n:{label} {"
    query += ", ".join([f"{key}: ${key}" for key in properties.keys()])
    query += "})"
    tx.run(query, **properties)

with driver.session() as session:
    with open('data.csv', mode='r') as file:
        reader = csv.DictReader(file)
        for row in reader:
            session.write_transaction(create_node, "LabNode", row)
```
```

## ##### Шаг 4: Выполнение операций с данными

### 4.1. \*\*MongoDB:\*\*

- Найдите документ по определённому критерию:

```
```python
result = collection.find_one({"field_name": "value"})
print(result)
```
```

- Обновите документ:

```
```python
collection.update_one({"field_name": "value"}, {"$set": {"field_name":
"new_value"}})
```
```

- Удалите документ:

```
```python
collection.delete_one({"field_name": "value"})
```
```

#### 4.2. **\*\*Redis:\*\***

- Получите данные по ключу:

```
```python
data = r.hgetall('row:1')
print(data)
```
```

- Обновите данные:

```
```python
r.hset('row:1', 'field_name', 'new_value')
```
```

- Удалите ключ:

```
```python
r.delete('row:1')
```
```

#### 4.3. **\*\*Neo4j:\*\***

- Найдите узел по определённому критерию:

```
```python
with driver.session() as session:
    result = session.run("MATCH (n:LabNode {field_name: $value}) RETURN n", value="value")
    for record in result:
        print(record['n'])
```
```

- Обновите узел:

```
```python
with driver.session() as session:
    session.run("MATCH (n:LabNode {field_name: $value}) SET n.field_name = $new_value", value="value", new_value="new_value")
```
```

- Удалите узел:

```
```python
with driver.session() as session:
    session.run("MATCH (n:LabNode {field_name: $value}) DELETE n", value="value")
```
```

#### Варианты для студентов:

Каждому студенту предоставляется CSV файл с уникальными данными, который он должен загрузить в MongoDB, Redis и Neo4j. Студент должен продемонстрировать выполнение базовых операций (вставка, выборка,

обновление, удаление) для каждой из СУБД. Ниже приводятся примеры тем, с которыми могут быть связаны данные в CSV:

1. Сотрудники компании и их контактные данные.
2. Товары интернет-магазина и их описание.
3. Книги и авторы в библиотеке.
4. Курсы и преподаватели в университете.
5. Пациенты и их медицинские карты.
6. Заказы и клиенты интернет-магазина.
7. Фильмы и режиссеры в кинобазе.
8. Услуги и клиенты сервисного центра.
9. Транспортные средства и их владельцы.
10. Продукты и их поставщики.
11. Сотрудники и их департаменты в организации.
12. Студенты и их курсы в образовательном учреждении.
13. Проекты и участники в IT компании.
14. Клиенты и их заказы в ресторане.
15. Маршруты и станции в транспортной сети.
16. Оборудование и его спецификации в производстве.
17. Музыкальные альбомы и артисты.
18. Туристические маршруты и гиды.
19. Программные продукты и их версии.
20. Клиенты и их счета в банке.
21. Бренды и их товары на рынке.
22. Квартиры и арендаторы в жилом комплексе.
23. Ивенты и участники в системе мероприятий.
24. Лекции и студенты в образовательной платформе.
25. Контракты и их условия для компании.

#### Отчет по лабораторной работе:

Отчет должен включать:

1. Описание загрузки данных в каждую из NoSQL СУБД.
2. Пример кода для работы с MongoDB, Redis и Neo4j.
3. Результаты выполнения операций с данными.
4. Выводы по работе с различными типами NoSQL баз данных.

#### Заключение:

Эта лабораторная работа позволит студентам ознакомиться с различными методами хранения данных в NoSQL системах на примере MongoDB, Redis и Neo4j. Студенты получают навыки работы с данными, загружаемыми из CSV файлов, а также научатся выполнять базовые операции в каждой из СУБД.