

Amaliy ish № 8

NoSQL bilan ishlash

Ishning maqsadi:

Katta ma'lumotlarni saqlash va tahlil qilish uchun hujjatga yo'naltirilgan ma'lumotlar bazalari (MongoDB) bilan ishlashni o'rganish.

Nazariy qism:

1. NoSQL nima?

NoSQL (Nafaqat SQL) - bu relyatsion jadvallardan foydalanmaydigan ma'lumotlarni boshqarishga yondashuv. Buning o'rniga, ma'lumotlar hujjatlar, grafiklar yoki kalit-qiymat juftliklari kabi turli formatlarda saqlanadi.

NoSQL ma'lumotlar bazalari quyidagilar uchun idealdir:

- Katta hajmdagi strukturalanmagan ma'lumotlarni saqlash.
- Gorizontaal masshtablash.
- Ma'lumotlarga tezkor kirish.

2. MongoDB nima?

MongoDB eng mashhur NoSQL ma'lumotlar bazalaridan biridir. U ma'lumotlarni JSON (BSON) hujjatlari ko'rinishida saqlaydi.

MongoDB ning asosiy xususiyatlari:

- Moslashuvchan ma'lumotlar sxemasi.
- Tez qidiruvlar uchun indeksni qo'llab-quvvatlash.
- Gorizontaal masshtablash imkoniyati.

3. Nima uchun sizga MongoDB kerak?

MongoDB quyidagilar uchun ishlatiladi:

- Katta hajmdagi strukturalanmagan ma'lumotlarni saqlash (masalan, jurnallar, ijtimoiy tarmoqlar, IoT).
- Tez-tez o'zgarib turadigan ma'lumotlar bilan ishlash.
- Yuqori unumdorlikni talab qiluvchi ilovalar.

4. MongoDB qanday o'rnatiladi?

- MongoDB-ni o'rnatish uchun uni rasmiy veb-saytdan yuklab oling: [MongoDB yuklab olish](#).

- Shu bilan bir qatorda, Docker-dan foydalanish mumkin:

```
docker run -d -p 27017:27017 --name mongoddb mongo
```

5. Python orqali MongoDB bilan qanday ishlash mumkin?

Pythonda MongoDB bilan ishlash uchun `pymongo` kutubxonasi ishlatiladi. Uni buyruq bilan o'rnatish:

- `pip install pymongo`

Amaliy qism:

1. MongoDB o'rnatilishi:

Agar Docker dan foydalanganda, MongoDB ni ishga tushirish:

```
docker run -d -p 27017:27017 --name mongoddb mongo
```

- Yoki MongoDB ni rasmiy veb-saytidan o'rnatish.

2. Ma'lumotlarni tayyorlash:

- Oldingi ishlarda yuklab olingan Titanic (`train.csv`) ma'lumotlar to'plamidan foydalaniladi.

3. Ma'lumotlar bilan ishlash:

- Yangi Python faylini yaratish (masalan, `amaliy_7.py`) va quyidagi amallarni bajarish:

```
import pandas as pd
from pymongo import MongoClient
```

```
# Ma'lumotlar yuklanishi
data = pd.read_csv('train.csv')
```

```
# Ma'lumotlarni lug'atlar ro'yxatiga aylantirish
data_dict = data.to_dict(orient='records')
```

```
# MongoDB ga ulanish
client = MongoClient('mongodb://localhost:27017/')
db = client['titanic_db'] # Ma'lumotlar bazasini yaratish
collection = db['passengers'] # To'plam yaratish
```

```
# MongoDB-ga ma'lumotlarni kiritish
collection.insert_many(data_dict)
```

```
# 1. 30 yoshdan oshgan barcha yo'lovchilar namunasi
query = {"Age": {"$gt": 30}}
results = collection.find(query)
print("30 yoshdan oshgan yo'lovchilar:")
for passenger in results:
    print(passenger)
```

```
query = {"Sex": "female", "Survived": 1}
count = collection.count_documents(query)
print (f"\nOmon qolgan ayollar n soni: {count}")
```

```
# 3. Chipta narxi bo'yicha ma'lumotlarni saralash
results = collection.find().sort("Fare", -1).limit(5)
print ("\nChiptalar narxi eng qimmat bo'lgan TOP-5 ta yo'lovchi:")
for passenger in results:
    print(passenger)
```

4. Dasturni ishga tushirish:

- Faylni saqlash va uni terminal orqali ishga tushirish

```
python amaliy_7.py
```

Natijada ko'rinadi:

- 30 yoshdan oshgan yo'lovchilar.
- Tirik qolgan ayollar soni.
- Eng qimmat chiptalar narxiga ega TOP-5 ta yo'lovchi.

Qo'shimcha materiallar:

- Rasmiy MongoDB hujjatlari: [MongoDB Documentation](#).
- PyMongo hujjatlari: [PyMongo hujjatlari](#).
- MongoDB qo'llanma: [MongoDB Tutorial](#).

Amaliy vazifa:

Vazifa: № 7

Titanic (`train.csv`) ma'lumotlar to'plami va MongoDB foydalanib quyidagi amallarni bajaring:

1. Omon qolgan barcha erkaklarni toping.
2. Har bir kabina sinfidagi (`Pclass`) yo'lovchilarning o'rtacha yoshini hisoblang.
3. birinchi toifali yo'lovchilar orasida maksimal chipta narxini toping (`Fare`).