

Mongo - Document DB

NoSQL baze podataka



Univerzitet u Novom Sadu
Fakultet tehničkih nauka

Mongo baza podataka

- ▶ *Zvanična dokumentacija*
- ▶ *Go driver*



Primer dokumenta

```
{
  _id: ObjectId("5099803df3f4948bd2f98391"),
  name: "Pera",
  surname: "Perić",
  birth: new Date('Jun 23, 1989'),
  phoneNumbers: ["555-333-222", "444-111-222"],
  address: {
    street: "Ulica Modene",
    city: "Novi Sad",
    country: "Serbia"
  },
  anamnesis: [
    {
      symptom: "runny nose",
      intensity: "high",
      startDate: new Date('Nov 3, 2022'),
    },
    {
      symptom: "sneezing",
      intensity: "moderate",
      startDate: new Date('Nov 3, 2022'),
    }
  ],
  therapy: [
    {
      name: "xyzal",
      price: 10
    }
  ]
}
```

MongoDB

- ▶ Source-available (*skoro open-source*)
- ▶ Document orijentisana baza podataka
- ▶ **Horizontalno skaliranje kroz *sharding***
- ▶ **Visoka dostupnost kroz *replikacije***
- ▶ Delimično razvijena podrška za transakcije
- ▶ **Raznovrsna podrška za pretrage i agregacije**
- ▶ Primarni i sekundarni indeksi
- ▶ Podrška za kreiranje šeme podataka

Osnovni pojmovi

Uporedni pregled pojmova u okviru MongoDB i SQL baza podataka:

- | | |
|-----------------|-----------------|
| ▶ Baza podataka | ▶ Baza podataka |
| ▶ Kolekcija | ▶ Tabela |
| ▶ Dokument | ▶ Torka |
| ▶ Polje | ▶ Kolona |

Napomena: mapiranje nije 1-1!

mongosh

Osnovne komande:

- ▶ **docker exec -it ime_kontejnera mongosh** - povezivanje na *mongosh* u okviru pokrenutog Docker kontejnera
- ▶ **ping** - testiranje konekcije
- ▶ **show dbs** - izlistavanje postojećih baza podataka
- ▶ **use ime_baze** - prebacivanje na bazu po imenu *ime_baze*
- ▶ **show collections** - izlistavanje postojećih kolekcija u okviru aktivne baze
- ▶ **db.dropDatabase()** - brisanje aktivne baze

Read

Sve navedene komande se izvršavaju nad kolekcijom - **db.ime_kolekcije.komanda**

- ▶ **find()** - dobavljanje svih dokumenata u okviru kolekcije
- ▶ **find(filter_objekat)** - dobavljanje svih dokumenata u okviru kolekcije koji zadovoljavaju *filter_objekat*
- ▶ **find(filter_objekat, select_objekat)** - slično; dobavlja samo polja koja su navedena u okviru *select_objekt-a*
- ▶ **findOne()** - slično; dobavlja samo prvi objekat
- ▶ **countDocuments(filter_objekat)** - prebrojava sve dokumente koji zadovoljavaju *filter_objekat*

Read

Paginacija i sortiranje:

- ▶ `sort(polje1: 1, polj2: -1)` - sortiranje po *polje1* ASC i po *polje2* DESC
- ▶ `limit(n)` - dobavljanje prvih *n* dokumenata
- ▶ `skip(n)` - preskakanje prvih *n* dokumenata

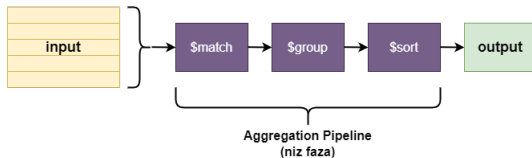
Kompleksni filter objekti

Sva navedena komande se mogu kombinovati u cilju kreiranja kompleksnih objekata za filtriranje

- ▶ **\$regex** - regularni izrazi: `{ name: { $regex: "[a-z]*" } }`
- ▶ **\$eq** - jednakost: `{ name: { $eq: "Pera" } }`
- ▶ **\$ne** - različito: `{ name: { $ne: "Pera" } }`
- ▶ **\$gt** ili **\$gte** - veće ili veće i jednako: `{ price: { $ge: 100 } }`
- ▶ **\$lt** ili **\$lte** - manje ili manje i jednako: `{ price: { $lte: 100 } }`
- ▶ **\$in** - pripadnost nizu vrednosti: `{ name: { $in: ["Pera", "Mika"] } }`
- ▶ **,** - logičko i: `{ name: "Pera", surname: "Perić" }`
- ▶ **\$or** - logičko ili: `{ $or: [{ name: "Pera" }, { age: 27 }] }`
- ▶ **\$exists** - proverava postojanja polja: `{ name: { $exists: true } }`
- ▶ **\$expr** - poređenje između polja: `{ $expr: { gt: ["$stanje", "$dug"] } }`

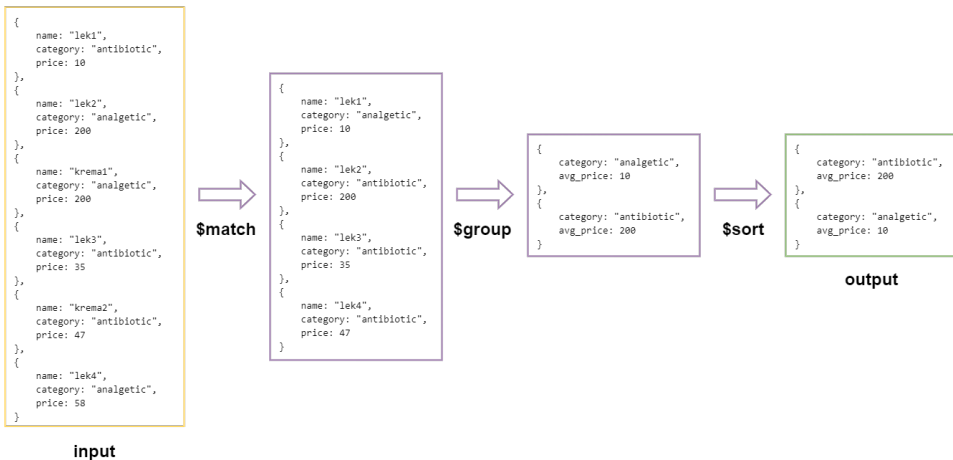
Agregacije

- ▶ Podrazumeva definisanje različitih faza (*stage*) koje se sukcesivno pozivaju nad kolekcijom dokumenata
- ▶ Izlaz iz prethodne faze predstavlja ulaz u narednu
- ▶ Primer agregacije:



```
db.collection.aggregate([
  { $match: { name: { $regex: "lek" } } },
  { $group: { _id: "$category", avg_price: { $avg: "$price" } } },
  { $sort: { avg_price: -1 } }
])
```

Agregacije - primer



Agregacije - faze

- ▶ **\$match** - faza filtriranja: `{ $match: { age: 2 } }`
- ▶ **\$group** - faza grupisanja (slično *GROUP BY* SQL naredbi): `{ $group: { _id: "$name", total: { $sum: "$price" } } }`
- ▶ **\$project** - faza projekcije (kreiranje izlaznog objekta): `{ $project: { "name": 1, "surname": 1 } }`
- ▶ **\$addFields** - faza dodavanja novog polja: `{ $addFields: { total: { $sum: "$price" } } }`
- ▶ **db.ime_kolekcije.aggregate(faze_objekat)** - pozivamo agregiranje nad datom kolekcijom

Partitionisanje

- ▶ **Sharding** - mehanizam podržan od strane MongoDB za partitionisanje podataka
- ▶ Deljenje podataka na više instanci MongoDB (najčešće na više mašina)
- ▶ Metod horizontalnog skaliranja
- ▶ Elementi arhitekture:
 - ▶ **Shard** - 1 (ili više) instanci MongoDB koja sadrži podatke
 - ▶ **Mongos** - ruter zahteva; usmerava zahtev do odgovarajuće instance koja sadrži tražene podatke
 - ▶ **Config servers** - konfiguracioni serveri; skladište konfiguraciju i metapodatke za ceo klaster
- ▶ Postoji više strategija na osnovu kojih se podaci mogu podeliti na *shard*-ove
- ▶ Više na: *link*

Particionisanje

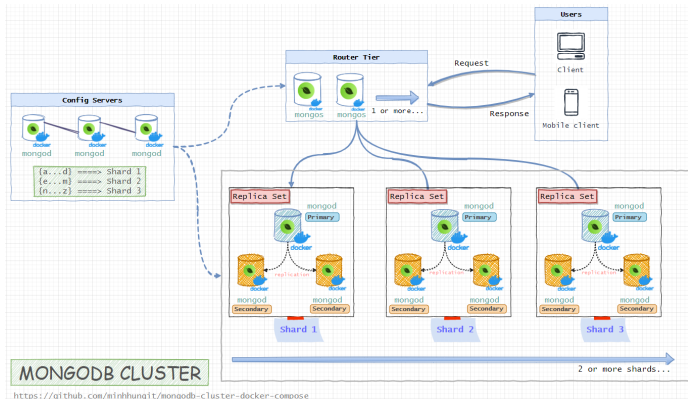
- ▶ Prednosti:
 - ▶ Rasterećenje jedne baze
 - ▶ Brži odgovori
 - ▶ Veća dostupnost (uz upotrebu replikacije)
- ▶ Mane:
 - ▶ Neophodno više resursa (računara)
 - ▶ Kompleksnija konfiguracija i struktura

Replikacija

- ▶ **Replica set** - grupa instanci MongoDB koje sadrže iste podatke
- ▶ Pruža redundandnost podataka i visoku dostupnost
- ▶ Postoji više strategija prilikom kreiranja replikacionih setova
 - ▶ Bitno nam je da znamo koja baza postaje "glavna" u slučaju da dođe do otkaza
 - ▶ Različite strategije replikacije podataka (npr. 1 primarna baza i 2 sekundarne)
 - ▶ Uvek imamo neparan broj instanci u okviru replikacionog seta (kako bismo imali većinsko glasanje)

Primer

Primer rešenja sa particionisanjem i replikacijom:



Preuzeto sa: *link*

Primer

U okviru primera *RestMongoAggregations* upotrebljeni su:

1. *Mongo* - baza podataka
2. *Docker* - kontejnerizacija rešenja (i "instalacija" baza)
3. *Go* - implementacija primera

Zadaci

- ▶ Proširiti servis tako da podržava računanje ukupne cene lekova:
 - ▶ za pacijenta na osnovu prosleđenog id-a
 - ▶ za svakog pacijenta
- ▶ **Bonus:** Smisliti i isprobati agregacije po izboru