

Napredne operacije i rad sa nizovima

Predmetni asistent:

Jovana Stojanović, dipl. inž. el. i rač.

jovana.stojanovic@akademijanis.edu.rs

Logički operatori – Moć izbora (**\$or**)

- Ako imamo više uslova. Ponekad nam podaci ne moraju ispuniti sve uslove, već samo jedan od ponuđenih.
 - { **\$or**: [{ uslov1 }, { uslov2 }, ...] }
- Primer:
 - Pronađi korisnike koji imaju 25 godina Ili se zovu "Marko":

```
db.users.find({
  $or: [
    { age: 25 },
    { name: "Marko" }
  ]
})
```

Logički operatori – Striktne uslovi (**\$and**)

- Spajanje uslova sa **\$and**. Iako MongoDB podrazumevano spaja uslove zarezom, **\$and** koristimo kada imamo više složenih uslova nad istim poljem.
 - **{ \$and: [{ uslov1 }, { uslov2 }] }**
- Primer:
 - Pronađi korisnike koji imaju više od 20 godina I manje od 30:

```
db.users.find({
  $and: [
    { age: { $gt: 20 } },
    { age: { $lt: 30 } }
  ]
})
```

Regularni izrazi – Pretraga po delu teksta (**\$regex**)

- Kada ne znamo tačan naziv, koristimo **\$regex** za pronalaženje obrazaca (pattern-a).

- Primeri:

- 1. Pronađi sva imena koja počinju na slovo "A":

```
db.users.find({ name: { $regex: "^A" } })
```

- 2. Pronađi sva imena koja sadrže "ana" (npr. Ana, Jovana, Dijana):

```
db.users.find({ name: { $regex: "ana", $options: "i" } })  
// "i" znači case-insensitive (nebitno veliko/malo slovo)
```

Projekcija – Biramo šta vidimo (Filtriranje polja)

- U realnim bazama dokumenti imaju na desetine polja. Projekcija nam omogućava da izvučemo samo ono što nam treba.
 - 1 = prikaži polje
 - 0 = sakrij polje
- Primer:
 - Vрати samo imena korisnika, sakrij godine i njihov sistemski ID:

```
db.users.find(  
  {}, // Prva zagrada je za filter (prazna znači "svi dokumenti")  
  { name: 1, _id: 0 } // Druga zagrada je projekcija  
)
```

Rezime operatora

Operator	Svrha	Primer
\$or	Bilo koji uslov tačan	<code>{ \$or: [...] }</code>
\$and	Svi uslovi tačni	<code>{ \$and: [...] }</code>
\$regex	Pretraga teksta	<code>{ name: /pattern/ }</code>
Projekcija	Biranje kolona	<code>{ name: 1, _id: 0 }</code>

Upiti nad nizovima – Operator **\$all**

- Standardni upit traži poklapanje, ali **\$all** osigurava da dokument sadrži sve navedene vrednosti, bez obzira na njihov redosled u nizu.
- Primer:
 - Pronađi kurseve koji imaju tagove "baze" i "web":

```
db.courses.find({
  tags: { $all: ["baze", "web"] }
})
```

Operatori \$size i \$elemMatch

- **\$size**: Pronalazi nizove koji imaju tačan broj elemenata.
- **\$elemMatch**: Koristi se kada niz sadrži objekte (dokumente) i želimo da jedan objekat ispunjava više uslova.

- Primeri:

- 1. Pronađi korisnike sa tačno 3 hobija:

```
db.users.find({ hobbies: { $size: 3 } })
```

- 2. Pronađi dokument gde je u nizu "ispiti" bar jedan predmet "Matematika" sa ocenom većom od 8:

```
db.students.find({  
  exams: { $elemMatch: { subject: "Matematika", grade: { $gt: 8 } } }  
})
```

Ažuriranje nizova – Dodavanje elemenata

- Kada menjamo postojeće dokumente, koristimo operatore za ažuriranje umesto da prepisujemo ceo niz.
 - **\$push**: Dodaje element na kraj niza (dozvoljava duplikate).
 - **\$addToSet**: Dodaje element samo ako on već ne postoji (ponaša se kao skup).
- Primeri:
 - Dodaj novu veštinu korisniku (ako već ne postoji)

```
db.users.updateOne(  
  { name: "Lazar" },  
  { $addToSet: { skills: "MongoDB" } }  
)
```

- Želimo da dodamo novu ocenu 10 u niz grades.

```
db.students.updateOne(  
  { name: "Marko" }, // Filter: nađi studenta koji se zove Marko  
  { $push: { grades: 10 } } // Akcija: dodaj 10 na kraj niza  
)
```

Ažuriranje nizova – Uklanjanje elemenata

- Operator **\$pull** uklanja sve instance vrednosti (ili vrednosti koje ispunjavaju uslov) iz postojećeg niza.
- Primer:
 - Ukloni tag "stari_kurs" iz svih dokumenata u kolekciji:

```
db.courses.updateMany(  
  {}, // Filter za sve dokumente  
  { $pull: { tags: "stari_kurs" } }  
)
```

Pregled operatora za nizove

Operator	Tip	Čemu služi?
<code>\$all</code>	Query	Sadrži sve elemente sa liste
<code>\$size</code>	Query	Tačna dužina niza
<code>\$elemMatch</code>	Query	Provera više uslova unutar niza objekata
<code>\$push</code>	Update	Dodaje element (duplikati OK)
<code>\$addToSet</code>	Update	Dodaje element (bez duplikata)
<code>\$pull</code>	Update	Briše element iz niza

Sortiranje rezultata

- Funkcija **.sort()** menja redosled kojim se dokumenti prikazuju.
 - 1 = Rastući redosled (A-Z, 0-9)
 - -1 = Opadajući redosled (Z-A, 9-0)

- Primer:
 - Prikaži sve proizvode od najjeftinijeg ka najskupljem:

```
db.proizvodi.find().sort({ cena: 1 })
```

Limitiranje i preskakanje

- **.limit(n)**: Ograničava rezultat na tačno n dokumenata.
- **.skip(n)**: Preskače prvih n dokumenata.