

## 2-10 Mongo upiti

### Tipovi podataka

Kao i u SQL bazama podataka, tako i MongoDB atributi u objektima imaju tipove. Neki od najčešćih tipova su: null, ObjectId, number, double, string, array, binData, bool, date, int, long. U MongoDB dokumentaciji može se pronaći paralelno poređenje sa SQL jezikom i naredbama kao i tipovima u MongoDB: <https://docs.mongodb.com/manual/reference/sql-comparison/>

Primjer unosa objekta sa različitim tipovima podataka:

```
db.studenti.insert({
  ime: 'Alisa',
  prezime: 'Alisovic',
  datum_rodj: ISODate('2005-03-01'),
  osobine: {
    godine: 15,
    visina: 160,
    tezina: 55
  },
  koeficijent: 3.37,
  grupa: 'D'
});
```

Ovo je primjer komplikovanijeg objekta. Za pretraživanje objekta spomenuta je `find({})` metoda. Tako se može pretraživati na nekoliko načina.

1. `db.studenti.find({ ime: 'Alisa' });`
2. `db.studenti.find({ prezime: 'Alisovic' });`
3. `db.studenti.find({ ime: 'Alisa', prezime: 'Alisovic' });`
4. `db.studenti.find({ datum_rodj: ISODate('2005-03-01 00:00:00.000Z') });`
5. `db.studenti.find({ 'osobine.godine': 15 });`

Ovdje se mogu vidjeti primjeri pretraživanja na različite načine, uključujući i pretraživanje po atributima pod objektima. Za izmjenu podataka, koristi se `update()` metoda:

```
db.studenti.update({ ime: 'Denis' },
  { $set:
    { prezime: 'Izmjena', godine: NumberInt(20) }
  });
```

Na prvom mjestu se naznači objekat za pretraživanje, a na drugom mjestu objekat sa specijalnim operaterom `$set`, i objektom sa atributima koje želimo primijeniti. Efekat ove komande je da

nepoklapajući atributi u originalnom objektu neće biti uklonjeni već će biti izmjenjeni samo podudarajući atributi ili dodani novi. Treba napomenuti da se `_id` atribut ne može mijenjati.

Recimo da želimo dodati niz objekata za studenta:

```
db.studenti.update({ ime: 'Denis' }, { $set: {
  hobiji: [
    { naziv: 'igre', godine: 3 },
    { naziv: 'ples', godine: 2 },
    { naziv: 'teretana', godine: 4 },
    { naziv: 'fudbal', godine: 8 }
  ]
}});
```

Dodan je novi niz, liste hobija za sve studente kojima je ime Denis. Na isti način je mogao biti potražen dokument sa specifičnim `_id`. Da se pretraže studenti po hobijima onda:

```
db.studenti.find({
  'hobiji.0.naziv': 'igre'
});
```

Ovaj upit će pronaći sve studente koji za hobije na prvom mjestu imaju igre.

## Operatori `$lt`, `$gt`, `$lte`, `$gte`

Ovi operatori omogućavaju pretraživanje poput manje (`$lt`), veće (`$gt`), manje ili jednako (`$lte`) i veće ili jednako (`$gte`). Ovi operatori se mogu primijeniti na sve tipove podataka. Npr.:

```
db.studenti.find({
  'godine': { $gte: 20 }
});
```

## Operatori `$in`

Ukoliko objekat ima niz brojeva poput:

```
db.studenti.insert({
  ime: 'Kenan', prezime: 'Kenanovic',
  koeficijenti: [ 3, 8, 10, 11, 18, 21, 27 ]
});
```

I želimo da pretražimo sve objekte koji u atributu koeficijent imaju brojeve 11 i 13, u te svrhe se koristi operator `$in`.

```
db.studenti.find({ koeficijenti: { $in: [11, 13] } });
```