Baze podataka

# Relacione baze podataka

**Relaciona baza podataka** je poseban tip baze podataka kod kojeg se organizacija podataka zasniva na relacionom modelu. Podaci se u ovakvim bazama organizuju u skup relacija između kojih se definišu određene veze. Relacija se definiše kao skup n-torki sa istim atributima, definisanih nad istim domenima iz kojih mogu da uzimaju vrednosti. U relacionim bazama podataka, svaka relacija mora da ima definisan ***primarni ključ*, koji predstavlja atribut pomoću kojeg se jedinstveno identifikuje svaka n-torka**. Relacija opciono može da poseduje i *spoljni ključ*, preko kojeg ostvaruje vezu sa drugim relacijama.

Upravljanje ovakvim bazama podataka se realizuje preko **sistema za upravljanje relacionim bazama podataka.**

Među najpopularnijim takvim sistemima danas su: Microsoft SQL Server, Oracle Database, MySQL i drugi. Većina tih sistema koristi upitni jezik **SQL** za manipulaciju podacima.

Relacione baze podataka karakteristike:

* strukturna jednostavnost
* formalno i strogo zasnivanje
* ekonomični upitni jezici
* razgraničenje nivoa predstavljanja podataka

# Nerelacione baze podataka

**NoSQL baze podataka** primenjuju drugačije mehanizme za smeštanje i povezivanje podataka za razliku od relacionih baza podataka. Ako nije potrebno da se skladište podaci u tabelama ili postoje veze koje se ne mogu predstaviti klasičnim SQL relacijama i pritom je potrebno da se brzo pristupi podacima onda se za to koristi NoSQL baza.

NoSQL baze podrazumevaju **nerelacione**, **distribuirane**, **baze** **otvorenog** **koda** i **horizontalno** **skalabilne**.

Karakteristike:

* Skalabilnost – mogućnost sistema da se nosi sa rastućom količinom podataka.
* Omogućavaju brzo indeksiranje i pretraživanje.
* Jednostavan spoljašnji interfejs ili protokol (za razliku od SQL-a).
* Sposobnost ponavljanja i distribucije podataka na vise servera.
* Sposobnost da dinamički dodaju nove atribute.
* Uglavnom podržavaju slabiji konkurentni model od ACID osobina transakcija (koje

su zastupljenje u SQL-u).

## SQLite

SQLite je ACID-kompatibilan ugrađen sistem za upravljanje bazama podataka sadržan u relativno maloj (~225 KB) C programskoj biblioteci. Izvorni kod za SQLite је u javnom vlašništvu.

Za razliku оd klijent-server sistema za upravljanje bazama podataka, jezgro SQLite-а nije samostalan proces sa kojim aplikacija komunicira.

Umesto toga, SQLite biblioteka је uvezena i postaje sastavni deo aplikacije.

Biblioteka može biti dinamično pozivana.

Aplikacija koristi SQLite funkcionalnost kroz proste funkcionalne pozive, koji smanjuju latenciju u pristupu bazama podataka kako su funkcionali pozivi unutar jednog procesa efikasniji od međuprocesne komunikacije.

*Primeri* kucanja upita u terminalu ili command prompt-u:

1. sqlite3 test.db "create table t1 (t1key INTEGER
2. PRIMARY KEY,data TEXT,num double,timeEnter DATE);"
3. sqlite3 test.db "select \* from t1 limit 2";

### Prednosti i mane SQLite u odnosu na druge relacione baze podataka

#### Prednosti:

1. **Čuva se kao jedan fajl** - pogodna za mobilne uređaje, televizore, kućne elektronske uređaje itd.
2. **Bolje perfomanse**
   * Brzo čitanje i upisivanje - 35% brže nego Fajl sistem
   * Učitavanje samo potrebnih podataka umesto celog fajla
   * Prilikom izmene određenog dela u fajlu, samo se taj deo čuva nazad
3. **Nije potrebna instalacija** – potrebno je samo skinuti neophodne biblioteke
4. **Pouzdanost**
   * Automatski čuva podatke
   * Kod u upitima je manji
5. **Portabilnost**
   * Može se koristiti na 32-bitnoj arhitekturi ili 64-bitnoj, kao i na big-and I little-endian arhitekturi
   * Dva odvojena procesa mogu istovremenu da upisuju i čitaju podatke iz baze
   * Može se koristiti sa svim programskim jezicima, bez problema sa kompatibilnošću
6. **Smanjuje cenu i kompleksnost programa**
   * Baza se lako proširuje novim tabelama i/ili kolonama

#### Mane:

1. Relacion baza – komplikovanije čuvanje veće količine podataka I pisanje samih upita, brzo raste u tom slučaju

## MongoDB

**MongoDB** je jedan od najpoznatijih predstavnika NoSQL baza podataka. To je "open-source" implementirana baza podataka, kodirana u programskom jeziku C++.

MongoDB čuva podatke kao **JSON** dokumente sa dinamičkim šemama i čini integraciju podataka u mnogim aplikacijama jednostavnijom i brzom.

Prihvaćena je kao backend softver brojnih značajnih veb sajtova i servisa, uključujući [eBay](https://www.ebay.com/), [Foursquare](https://foursquare.com/) i [New York Times](https://www.nytimes.com/).

Karakteristie:

* Podržava automatsku distribuciju dokumenata među serverima.
* Ne obezbeđuje globalnu konzistenstnost, samo lokalnu.
* Obezbeđuje automičke operacije direktno nad poljima u dokumentima.
* *GridFS* - čuva fajlove bilo koje veličine bez komplikacija. Umesto čuvanja fajla u pojedinačnom dokumentu, GridFS deli fajl na delove, ili blokove i čuva svaki od tih blokova kao poseban document
* Automatsko skaliranje - Horizontalno skaliranje bez ugrožavanja funkcionalnosti. Horizontalno skaliranje distribuira jedan logički sistem baze podataka izmedju skupa mašina

|  |  |
| --- | --- |
| SQL | MongoDB |
| **CREATE TABLE student (**  **mbr INT,**  **prezime CHAR (50)**  **)** | Implicitno – unosom prvog dokumenta u kolekciju. **Može I eksplicitno:**  **db.createCollection(“student”)** |
| INSERT INTO student ( **100,**  **‘Šoštaković’);** | db.student.insert( **{mbr:100, prezime: ‘Šostaković’}**  **)** |
| SELECT \* FROM student; | db.student.find(); |
| SELECT prezime **FROM student**  **WHERE mbr = 200**  **ORDER BY prezime;** | db.student.find( **{mbr:100},**  **{prezime:1}**  **).sort({prezime:1});** |

Nakon održanog sastanka tim Summanus je doneo sledeću odluku:

*Za izradu web aplikacije Olimijada koristiće se* ***SQLite*** *baza podataka zbog gore navedenih prednosti.*

*Ukoliko bude potrebe za skladištenje veće količine podatka koristeće se MongoDB baza podataka.*