

## LIMBAJE PENTRU BAZE DE DATE

Ana CASTRAVEȚ

Departamentul Ingineria Software și Automatică, grupa TI-181 F/R, Facultatea Calculatoare, Informatică și Microelectronică, Universitatea Tehnică a Moldovei, Chișinău, Republica Moldova

Autorul corespondent: Ana CASTRAVEȚ, e-mail: [ana.bolocan@isa.utm.md](mailto:ana.bolocan@isa.utm.md)

Conducător științific: Dorian SARANCIUC, DISA, FCIM, UTM

**Rezumat.** Acest rezumat se referă la limbajele utilizate pentru a interoga bazele de date. Limbajul SQL (Structured Query Language) este cel mai utilizat limbaj în bazele de date relaționale și este utilizat pentru a crea, a modifica și a interoga bazele de date. Limbajul PL/SQL (Procedural Language/Structured Query Language) este un limbaj de programare utilizat în bazele de date Oracle. Bazele de date NoSQL utilizează, de asemenea, limbaje de interogare specifice, cum ar fi MongoDB Query Language și Cassandra Query Language. Toate aceste limbaje sunt esențiale în gestionarea datelor și permit utilizatorilor să efectueze operații de tipul select, insert, update și delete.

**Cuvinte cheie:** SQL (Structured Query Language), interogare, programare, funcții, proceduri, operațiuni.

### Introducere

Limbajele pentru bazele de date sunt utilizate pentru a comunica cu sistemul de gestiune a bazei de date (DBMS) și pentru a manipula datele stocate în aceasta.

SQL este unul dintre cele mai populare și utilizate limbaje pentru baze de date. Acesta este utilizat pentru a crea, modifica și interoga bazele de date. SQL este un limbaj declarativ, ceea ce înseamnă că utilizatorii trebuie doar să specifice ceea ce doresc să realizeze, iar sistemul de gestiune a bazei de date va determina cum să îndeplinească aceste cerințe [1].

PL/SQL și T-SQL sunt limbaje procedurale care permit utilizatorilor să creeze proceduri stocate și funcții pentru a manipula datele. Acestea oferă utilizatorilor un grad mai mare de control și flexibilitate în manipularea datelor. În plus, există și alte limbaje specializate pentru bazele de date, cum ar fi XML, care este utilizat pentru a gestiona datele în format XML, și RDF, care este utilizat pentru a gestiona datele în format PDF.

În general, limbajele pentru baze de date sunt esențiale pentru a lucra cu datele stocate în bazele de date și pentru a realiza operațiuni precum crearea, actualizarea, ștergerea și interogarea acestora. Înțelegerea și utilizarea acestor limbaje este importantă pentru oricine dorește să lucreze cu baze de date.

### Ce este SQL și cum funcționează?

SQL (Structured Query Language) este un limbaj de programare utilizat pentru gestionarea și manipularea datelor în bazele de date relaționale. Acesta permite utilizatorilor să creeze, să modifice și să întrebe bazele de date pentru a obține informațiile dorite [2].

### Ce este NoSQL și care sunt principalele sale caracteristici?

NoSQL (Not only SQL) este un termen generic care se referă la bazele de date non-relaționale. Acestea sunt proiectate pentru a gestiona cantități mari de date, care pot fi structurate sau nestructurate. Principalele caracteristici ale bazelor de date NoSQL includ scalabilitatea orizontală, toleranța la eșecuri, flexibilitatea structurii datelor și performanța.

### Ce este MongoDB și cum funcționează ?

MongoDB este o bază de date NoSQL care utilizează un model de stocare documental. Acesta stochează datele în documente JSON (JavaScript Object Notation) și utilizează o abordare distribuită pentru a asigura scalabilitatea orizontală [3].

### **Ce este MySQL și cum funcționează?**

MySQL este o bază de date relațională, care utilizează un limbaj de interogare SQL pentru a gestiona și manipula datele.

### **Ce sunt limbajele de interogare și care este rolul lor în gestionarea bazei de date?**

Limbajele de interogare sunt limbaje de programare utilizate pentru a interoga bazele de date și a obține informații specifice. Acestea permit utilizatorilor să creeze interogări complexe pentru a căuta, sorta și filtra datele în baza de date. Prin utilizarea limbajelor de interogare, utilizatorii pot accesa și manipula datele în mod eficient și precis.

Limbajele de bază de date oferă suport pentru proceduri stocate, inclusiv SQL și PL/SQL pentru Oracle, Transact-SQL pentru Microsoft SQL Server, PL/pgSQL pentru PostgreSQL și PL/SQL pentru MySQL [4].

Procedura stocate pot fi utilizate pentru a realiza o varietate de sarcini, cum ar fi:

1. Validarea datelor introduse în baza de date;
2. Crearea și gestionarea tabelor și a altor obiecte de bază de date;
3. Efectuarea de interogări complexe și raportări de date;
4. Actualizarea datelor existente în baza de date;
5. Executarea de operații de sincronizare și de replicare a datelor;
6. Gestionarea tranzacțiilor și a erorilor de bază de date;

Un avantaj important al procedurilor stocate este că acestea pot fi optimizate, ceea ce duce la o performanță mai bună și la o reducere a traficului de rețea. De asemenea, procedurile stocate pot fi securizate și controlate pentru a limita accesul la date și pentru a evita erorile de utilizare a bazei de date [5].

În general, utilizarea procedurilor stocate poate duce la o îmbunătățire semnificativă a performanței aplicațiilor de bază de date și la o reducere a complexității codului.

### **Concluzii**

În concluzie, limbajele de bază de date sunt esențiale pentru gestionarea și organizarea datelor într-o manieră eficientă și ușor de accesat. Acestea permit programatorilor să creeze, să acceseze și să manipuleze datele dintr-o bază de date, oferind instrumentele necesare pentru a efectua operațiuni complexe și a asigura integritatea datelor.

Limbajele de bază de date diferă de la o platformă la alta, dar cele mai utilizate sunt SQL și NoSQL. SQL este un limbaj relațional utilizat în bazele de date relaționale, în timp ce NoSQL este utilizat în bazele de date non-relaționale. În plus, există și alte limbaje de bază de date, cum ar fi PL/SQL, T-SQL și PL/pgSQL, care sunt utilizate pentru a crea proceduri stocate și funcții.

Procedurile stocate sunt de asemenea foarte importante în gestionarea datelor într-o bază de date și permit programatorilor să creeze funcții personalizate care pot fi apelate pentru a efectua o varietate de sarcini.

În general, utilizarea limbajelor de bază de date și a procedurilor stocate poate duce la o gestionare mai eficientă și securizată a datelor și poate îmbunătăți performanța aplicațiilor care accesează bazele de date.

### **Referințe**

1. Documentația oficială SQL [online]. [accesat 20.11.2022]. Disponibil: -[SQL Tutorial \(w3schools.com\)](https://www.w3schools.com/sql/);
2. Documentația oficială NoSQL [online]. [accesat 20.11.2022]. Disponibil: -[What Is NoSQL? NoSQL Databases Explained | MongoDB](https://www.mongodb.com/docs/guides/introduction/);
3. Documentația oficială Oracle PL/SQL [online]. [accesat 20.11.2022]. Disponibil: -[Oracle Database Database PL/SQL Language Reference, 19c](https://docs.oracle.com/en/database/oracle/oracle-database/19/plsql/);
4. Documentația oficială Microsoft T-SQL [online]. [accesat 20.11.2022]. Disponibil: -[Transact-SQL reference \(Database Engine\) - SQL Server | Microsoft Learn](https://docs.microsoft.com/en-us/sql/t-sql/language-reference/);
5. Documentația oficială PostgreSQL PL/pgSQL [online]. [accesat 20.11.2022]. Disponibil: -[PostgreSQL: Documentation: 15: Chapter 43. PL/pgSQL – SQL Procedural Language](https://www.postgresql.org/docs/15/plpgsql.html).