

Tema 4: Schema bazei de date în limbajul SQL

Schema bazei de date reprezintă structura logică a bazei de date. Ea descrie **cum sunt organizate datele**, nu datele propriu-zise.

Ca exemplu avem: dacă o comparăm cu *planul unei clădiri*, nu cu locatarii din ea.

O schemă include:

- tabelele;
- câmpurile (coloanele) tabelor;
- tipurile de date;
- chei primare;
- chei străine;
- relațiile dintre tabele;
- constrângeri (reguli).

În limbajul SQL, **schema** este și un obiect logic care:

- grupează tabelele și alte obiecte;
- ajută la organizarea bazei de date;
- controlează accesul utilizatorilor.

Exemple de scheme frecvente:

- `dbo` (implicită în SQL Server)
- `public` (în PostgreSQL)

Crearea unei scheme în SQL

```
CREATE SCHEMA scoala;
```

*Creează o schemă numită **scoala**, în care vor fi plasate tabelele.*

Crearea unui tabel într-o schemă

```
CREATE TABLE scoala.Elevi (  
    IdElev INT PRIMARY KEY,  
    Nume VARCHAR(50),  
    DataNasterii DATE,  
    Media FLOAT  
);
```

Deci, `tabelul` este identificat complet prin **schemă + nume tabel**. Relația dintre tabele în cadrul schemei este:

Exemplu:

- Elevi
- Clase

```
CREATE TABLE scoala.Clase (
    Id_Clasa INT PRIMARY KEY,
    Denumire VARCHAR(20)
);

CREATE TABLE scoala.Elevi (
    Id_Elev INT PRIMARY KEY,
    Nume VARCHAR(50),
    IdClasa INT,
    FOREIGN KEY (IdClasa) REFERENCES scoala.Clase(IdClasa)
);
```

✓ Aici schema conține **relații bine definite** între tabele.

Avantajele utilizării schemelor

- organizare clară a bazei de date;
 - evitarea conflictelor de nume;
 - securitate mai bună;
 - întreținere ușoară.
- ✓ Schema = structura bazei de date
 - ✓ Tabelele se creează în interiorul unei scheme
 - ✓ Schema ajută la organizare și securitate
 - ✓ Relațiile se definesc între tabele din schemă

Este obligatoriu să creăm conturi de utilizatori?

➤ **Nu este obligatoriu** să creăm conturi de utilizatori pentru a crea scheme.

✓ O **schemă** poate fi creată de un utilizator care **există deja** și are drepturile necesare.

Conturile de utilizatori sunt necesare **doar dacă**:

- dorim ca schema să aparțină unui anumit utilizator;
- vrem să controlăm strict accesul la obiectele din schemă;
- avem mai mulți utilizatori care lucrează în aceeași bază de date.

```
CREATE USER profesor IDENTIFIED BY 'parola';
CREATE SCHEMA scoala AUTHORIZATION profesor;
```

O schemă poate fi creată de:

- administratorul bazei de date;
- un utilizator existent cu permisiunea `CREATE SCHEMA`.

Sarcină individuală: Creează o schemă SQL cu denumirea **biblioteca**.

În cadrul schemei **biblioteca**, creează următoarele tabele, fiecare tabel să fie înregistrat 3 înregistrări:

a) Carti

- CarteID – cheie primară
- Titlu – NOT NULL
- Autor – NOT NULL
- AnPublicare
- Editura

b) Cititori

- CititorID – cheie primară
- Nume – NOT NULL
- Prenume – NOT NULL
- Clasa
- Email

c) Imprumuturi

- ImprumutID – cheie primară
- DataImprumut – NOT NULL
- DataReturnare
- CarteID – cheie străină
- CititorID – cheie străină