



# Sisteme de gestiune a bazelor de date


## Capitolul 1

**Tipuri de date si operații specifice  
acestora**

# Recapitulare


- Toate programele prezentate în anii anteriori au ca scop prelucrarea **informatiilor**.
- Informatia prelucrată poate fi sub forma de texte, numele, imagini, sunete.
- Informatia este păstrată sub formă de **date**.


- Calculatoarele nu pot procesa decât siruri binare de 0 si 1.
- Fiecărui tip de informatie îi corespunde un anumit mod de păstrare în memorie, adică un anumit **tip de dată**.
- Tipul datei determină operatiile care se pot executa cu acele date.
- **Exemplu:** *un sir de caractere se reprezintă sub forma unui grup de 8 cifre binare corespunzătoare codului ASCII si asupra acestui tip de dată se poate aplica operatorul de concatenare si operatorii relationali.*


- 
- Între datele prelucrate de un program există **relatii**.
  - Datele se pot organiza în colecții de date precum: **fisierul de date** și **baza de date**.
  - Colecțiile de date permit stocarea unor cantități mari de informații și prelucrarea lor în vederea obținerii de noi informații.
  - Pentru exploatarea datelor organizate sub formă de baze de date se folosește un software numit **sistem de gestiune a bazelor de date**.

# 1. Datele


- Datele sunt folosite pentru descrierea evenimentelor produse în cadrul unui proces.
- Prin prelucrarea lor cu ajutorul unui sistem de calcul se obțin informațiile.
- **Data reprezintă un model de reprezentare a informației, accesibil calculatorului, cu care se poate opera pentru a obține noi informații.**


- 
- **Informația** este un mesaj care face cunoscut un anumit eveniment și are caracterul noutății. Informațiile sunt interpretate de către oameni.
  - **Data** este reprezentarea informației în interiorul calculatorului. Calculatorul nu înțelege conținutul acestor date, el numai le prelucrează prin operații specifice fiecărui tip de dată.
  - În urma prelucrării **datelor**, calculatorul poate furniza omului **informații**.

- 
- Datele sunt înregistrate pe **suporturile de memorare** (memoria internă sau memoria externă) într-un anumit **format** care determină **semnificația datei**.
  - În funcție de formatul ales pentru o dată, sistemul va *interpreta* valoarea înregistrată pe suportul de memorare și va permite *folosirea anumitor operatori*.

- 
- Prelucrarea datelor se face **numai în memoria internă** a calculatorului.
  - Datele care urmează să fie prelucrate se numesc **date de intrare**. Ele sunt citite de la un dispozitiv de intrare (tastatură, scanner, creion optic etc.) sau dintr-o memorie externă, unde se pot găsi într-o colecție de date (fișier sau bază de date).
  - În urma prelucrărilor se obțin rezultatele, adică **datele de ieșire**. Pentru a putea fi vizualizate de către om, ele vor fi scrise la un dispozitiv de ieșire (ecran, imprimantă etc). Ele mai pot fi scrise într-o colecție de date în memoria externă, pentru a fi prelucrate ulterior ca date de intrare.



- 
- **Dispozitivul standard de intrare** (dispozitivul de la care calculatorul citește implicit datele de intrare) este **tastatura**, iar **dispozitivul standard de ieșire** (dispozitivul la care calculatorul scrie implicit datele de ieșire) este **ecranul**.

- 
- Cantitatea de date care se prelucrează pentru a obține informații diferă de la o aplicație la alta. Dacă se prelucrează foarte puține date, se vor folosi numai **date elementare**, iar dacă volumul datelor prelucrate este foarte mare, ele vor fi organizate în colecții de date, numite **structuri de date**.


# 1.1. Datele elementare


- **Data elementară** reprezintă o entitate indivizibilă atât în raport cu informația pe care o reprezintă, cât și din punct de vedere al procesorului care o prelucrează (programul sau unitatea centrală).


- 
- **Data elementară** poate fi definită prin tripletul:

data elementară = (identificator, atribute,  
valoare)


- **Identificatorul datei** este numele atribuit datei pentru a o putea deosebi de alte date în timpul procesului de prelucrare.
- Atunci când stabilim un nume pentru o dată, trebuie să respectate **constrângerile** sistemului în care se lucrează:
  - numărul maxim de caractere
  - caracterele acceptate
  - caracterele cu care poate să înceapă numele
- Prin identificatorul datei, utilizatorul are acces la **zona de memorie** internă în care este păstrată data, deoarece la declararea datelor sistemul își creează o tabelă de corespondență între identificatorul datei și adresa de memorie la care se găsește data.

- 
- **Valoarea datei** este conținutul zonei de memorie în care este păstrată data. Această valoare se poate modifica sau nu în timpul procesului de prelucrare.
  - În funcție de modul în care evoluează valoarea datei, există:
    - date constante (constante)
    - date variabile (variabile de memorie).


- 
- Atributele datei sunt proprietăți ale datei care determină modul în care sistemul tratează data.
  - Cel mai important atribut al unei date este **tipul datei**.


- 
- **Tipul datei** definește apartenența datei la o clasă de date care au aceeași reprezentare în memoria internă și care permit folosirea aceluiași operatori.
  - Tipul datei este definit de **domeniul de definiție al datei** și de **mulțimea operatorilor** care se pot aplica pe acele date.
  - Tipul datei determină **dimensiunea zonei de memorie** alocate datei și metoda folosită pentru **codificarea în binar** a datei.





- 
- **Tipul numeric** a fost implementat pentru reprezentarea numerelor (întregi sau reale).
  - Asupra datelor de tip numeric se pot aplica numai operatori matematici și relaționali.
  - **Constante numerice:**
    - 2.35
    - -234.5
    - 35
    - -75
    - -0.25
    - 3.14.


- **Tipul șir de caractere** a fost implementat pentru reprezentarea unei mulțimi ordonate de caractere (texte).
- Asupra datelor de acest tip se pot aplica numai **operatorii relaționali** și **operatorul de concatenare**.
- **Lungimea șirului** de caractere reprezintă numărul de caractere din șir.
- **Șirul vid** este șirul de lungime zero.
- **Subșirul** este o porțiune dintr-un șir care se poate identifica prin poziția din care începe și prin lungime.
- **Constante șiruri de caractere:**
  - "Ana",
  - 'Maria',
  - '10''',
  - "10"


- 
- **Tipul logic** a fost implementat pentru reprezentarea datelor care pot lua doar două valori (adevărat sau fals).
  - Asupra datelor de tip logic se pot aplica numai **operatori logici**.
  - **Constante logice:**
    - .T. (adevărat)
    - .F. (fals).

- 
- **Operatorii** sunt caractere speciale (+, \*, /, >, >=) sau cuvinte rezervate ale sistemului (mod, div, and, not).
  - Ei pot fi folosiți pentru a reprezenta operațiile din cadrul unei expresii.
  - Pot fi aplicați numai pe anumite tipuri de operanzi, producând rezultate de un anumit tip.


- 
- **Operatorul de atribuire** este = și poate fi folosit pentru a atribui o valoare unei date.
  - De exemplu,  $n = 100 * 2 + 10$ , înseamnă că datei  $n$  i se atribuie valoarea 210 obținută în urma evaluării expresiei  $100 * 2 + 10$ .


- 
- Operatorii matematici sunt  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $\%$ ,  $^$ .
  - Ei se aplică pe date numerice și furnizează rezultate de tip numeric.
  - Se folosesc pentru a executa operațiile matematice de bază: adunare, scădere, înmulțire, împărțire, ridicare la putere, restul și catul unei împărțiri a două numere întregi.

- 
- **Operatorii relaționali** sunt  $>$ ,  $>=$ ,  $<$ ,  $<=$ ,  $=$ ,  $<>$ .
  - Ei se aplică pe date de tip numeric sau de tip șir de caractere și furnizează rezultate de tip logic.
  - Se folosesc pentru a compara două numere sau două șiruri de caractere.


- 
- **Operatorul de concatenare este +.** El se aplică pe date de tip șir de caractere și furnizează un rezultat de tip șir de caractere.
  - Este folosit pentru a reuni două sau mai multe șiruri de caractere într-un singur șir de caractere.



- 
- **Operatorii logici** sunt **not, and. or.** Ei se aplică pe date de tip logic și furnizează rezultate de tip logic.
  - Se folosesc pentru a exprima diferite condiții compuse pe care trebuie să le îndeplinească anumite date.

- 
- **Expresia** este o combinație validă de operatori și operanzi. Operanzii pot fi nume de date, constante sau funcții. Combinația este validă numai dacă fiecare operator leagă doi operanzi care permit folosirea lui.

- **Precedența operatorilor** reprezintă ordinea în care sunt evaluați operatorii dintr-o expresie.
- De regulă, această ordine este:
  - funcțiile,
  - operatorii matematici,
  - operatorii de concatenare,
  - operatorii relaționali
  - operatorii logici.
- Operatorii matematici și logici au niveluri de prioritate diferită. De exemplu, pentru operatorii matematici există trei niveluri de priorități:
  - primul nivel conține ridicarea la putere ( $**$  sau  $^$ ),
  - al doilea nivel conține înmulțirile și împărțirile ( $*$ ,  $/$ ,  $\text{mod}$ ,  $\text{div}$ )
  - al treilea nivel conține adunarea și scăderea ( $+$ ,  $-$ ).
- Pentru niveluri de prioritate diferită, evaluarea se face în ordine: primul nivel, al doilea și al treilea.
- Operatorii relaționali au același nivel de prioritate.
- Pentru același nivel de prioritate, evaluarea operatorilor se face în ordinea în care apar în expresie, de la stânga la dreapta.
- Puteți schimba ordinea de evaluare folosind parantezele

- 
- **Funcția** este o prelucrare definită de autorii sistemului printr-un program, care în urma execuției furnizează o valoare chiar prin numele său.
  - Funcția poate fi folosită în cadrul unei expresii ca un operand, deoarece numele său reprezintă valoarea obținută în urma executării programului.
  - Astfel, pentru calculul radicalului dintr-un număr se folosește funcția  $\text{sqrt}(x)$ .


- În următorul tabel vă sunt prezentate toate combinațiile posibile de operatori și operanzi, folosind tipurile de date precizate:

Tipul operanzilor	Tipul operatorului	Tipul rezultatului
numeric	matematic	numeric
numeric	relațional	logic
sir de caractere	concatenare	sir de caractere
sir de caractere	relațional	logic
logic	logic	logic


## 1.2. Structurile de date


- Memoria internă a calculatorului este organizată ca un ansamblu de celule separate care au adrese consecutive.
- Într-o astfel de celulă sau într-un grup de celule adiacente se poate memora o dată elementară.
- În același mod este organizată și memoria externă: un ansamblu de locații de memorare numite sectoare, care au adrese consecutive.


- În cadrul unei aplicații pot să apară multe date de același tip.
- **De exemplu**, o firmă vrea să prelucreze cu ajutorul calculatorului situația vânzărilor. Informația prelucrată va fi organizată logic într-un tabel în care fiecare produs reprezintă un rând și fiecare zi dintr-o anumită perioadă (săptămână, lună, an etc.) reprezintă o coloană. Acest mod de organizare permite o analiză rapidă a informației și ușurează munca de obținere a informațiilor statistice (volumul vânzărilor dintr-o zi, volumul vânzărilor pentru un produs într-o anumită perioadă, cel mai bine vândut produs într-o anumită perioadă etc.).


- 
- Organizarea logică a informațiilor în exteriorul calculatorului trebuie să se regăsească și în interiorul lui, în modul în care sunt organizate datele sub formă de colecții de date care reprezintă informațiile.
  - Utilizatorul unei aplicații va gândi natural, fără să fie preocupat de modul în care sunt organizate efectiv datele în memoria calculatorului.
  - **Colecția de date** sau **structura de date** reprezintă o metodă de aranjare a datelor care sunt dependente unele de altele în cadrul unei aplicații.




- 
- **Structura de date** este o colecție de date între care se definește un anumit tip de relație ce determină metoda de localizare și de prelucrare a lor.

- 
- **Orice structură de date este o entitate de sine stătătoare**, care poate fi **identificată printr-un nume** și ale cărei componente își mențin attributele.
  - Fiecărui tip de structură de date îi este specific un anumit mecanism de identificare și de selecție a componentelor colecției de date.
  - Componentele structurii de date pot fi identificate fie prin **numele componentei**, fie prin **poziția** pe care o ocupă în cadrul structurii
  - **De exemplu**, în cazul unei înregistrări, elementele structurii sunt câmpurile și identificarea lor se face după numele câmpului, iar în cazul fișierelor de date elementele structurii sunt înregistrările și identificarea lor se face după numărul de ordine al înregistrării.
  - Există criterii care se pot folosi în clasificarea structurilor de date și care stabilesc ce operații se pot executa cu structurile de date.

- 
- **Tipul de structură de date** definește apartenența structurii de date la o anumită familie de structuri de date, cărora le corespunde același mod de organizare logică, același model *ae* reprezentare fizică și care pot fi supuse acelorași operații

- 
- **Tipurile de structuri de date folosite sunt:**
    - Tabloul de memorie
    - Înregistrarea
    - Fisierul de date
    - Baza de date
  - Asupra structurilor de date pot fi executate mai multe **operații** care pot afecta valorile componentelor structurii și/sau structura de date

- Operații care pot afecta structura bazei de date:
  - **crearea** (*create*) = realizarea structurii de date în forma inițială, pe suportul de memorare folosit;
  - **consultarea** = realizarea accesului la elementele structurii în vederea prelucrării și extragerii de informații;
  - **actualizarea** (*update*) = schimbarea stării structurii pentru a reflecta corect valoarea componentelor la un moment dat; operațiile de actualizare sunt:
    - **adăugarea** (*append*) de noi componente,
    - **ștergerea** (*delete*) unor componente,
    - **modificarea** (*edit*) valorii unor componente.
  - **sortarea** (*sort*) rearanjarea componentelor structurii în funcție de anumite criterii de ordonare aplicate valorii unor componente;


- 
- **copierea** (*copy*) realizarea unei imagini identice a structurii de date pe același suport de memorare sau pe un alt suport;
  - **mutarea** (*move*) transferarea structurii de date pe același suport de memorare la o altă adresă sau pe un alt suport;
  - **divizarea** (*split*) realizarea a două sau mai multe structuri dintr-o structură de bază;
  - **concatenarea** (*merge* sau *concatenation*) realizarea unei structuri prin reunirea a două sau mai multe structuri de date de același tip;
  - **redenumirea** (*rename*) schimbarea numelui structurii;
  - **ștergerea** (*erase*) distrugerea structurii de date.

În funcție de:

- **tipul componentelor structurii**, structurile de date pot fi **omogene** (componentele structurii sunt de același tip) sau **neomogene** (componentele structurii nu sunt de același tip).
- **tipul de memorie** în care sunt create, structurile de date pot fi **interne** (în memoria internă) sau **externe** (în memoria externă).
- **timpul de utilizare**, structurile de date pot fi **temporare** sau **permanente** (pot fi organizate numai în memoria externă).
- **modul de localizare a componentelor**, structurile de date pot fi cu **acces direct** (o componentă poate fi localizată fără să se țină cont de componentele care o preced) sau cu **acces secvențial** (o componentă poate fi localizată numai dacă se parcurg toate componentele care o preced).
- **reprezentarea pe suport**, structurile de date pot fi **explicite** (pe suport se înregistrează informații suplimentare necesare pentru localizarea componentelor) sau **implicite** (pe suport nu se înregistrează informații suplimentare).
- **stabilitate**, structurile de date pot fi **dinamice** (în timpul procesului de prelucrare își modifică numărul de componente și relațiile dintre ele) sau **statice** (în timpul procesului de prelucrare nu își modifică numărul de componente și relațiile dintre ele).

- **Tabloul de memorie** (*array*) este o **structură de date internă** formată dintr-o **mulțime de elemente**, ordonate după un ansamblu de indici (tabloul este format din elementul 1, elementul 2, elementul 3, .., elementul  $i$ ,.... iar ansamblul de indici este: 1,2,3,...,  $i$ ,...).
- **Dimensiunea** tabloului de memorie este dată de numărul de ansambluri de indici folosiți pentru ordonare. Dacă se folosește un singur ansamblu de indici, tabloul este **unidimensional** (cu o singură dimensiune) și se mai numește și **vector**. Dacă se folosesc două ansambluri de indici, tabloul este **bidimensional** (cu două dimensiuni) și se mai numește și **matrice**.
- Tabloul de memorie se identifică după un **nume** (de exemplu *alfa*), iar fiecare element al său după numele tabloului și numărul de ordine.
- Numărul de ordine este furnizat de un indice pentru vector (de exemplu *alfafi*), indicele fiind  $i$ ) sau de doi indici pentru matrice (de exemplu *alfa(ij)*, indicii fiind  $i$  și  $j$ ).
- În memoria internă, tabloul de memorie este o zonă continuă în care pot fi memorate mai multe date elementare.
- Zonei de memorie  $i$  se atribuie un nume (numele tabloului). Datele din tablou vor putea fi tratate ca un tot unitar sau ca date elementare independente.




- 
- Pentru a defini un tablou de memorie trebuie precizat: **numele tabloului, dimensiunea tabloului și numărul de elemente** pe fiecare dimensiune ( $n$  pentru vector sau  $n$  și  $m$  pentru matrice).
  - Tablourile de memorie sunt structuri omogene sau neomogene (în funcție de cum sunt implementate în sistem), interne, temporare, cu acces direct, implicite și statice (după definirea tabloului, nu mai poate fi modificată lungimea structurii).
  - Asupra structurii de date de tip tablou de memorie se pot executa următoarele operații:
    - crearea,
    - Consultarea
    - Actualizarea
    - Copierea.
  - Adăugarea sau inserarea de elemente se face fără să se modifice lungimea tabloului),
  - Sortarea presupune crearea unui nou tablou de memorie, cu elementele ordonate conform criteriului, sau rearanjarea elementelor în cadrul structurii.
  - Concatenarea și divizarea presupun crearea unor noi structuri de date, obținute prin reuniunea, respectiv prin divizarea structurilor sursă.


- Fișierul de date (*data file*) este o structură de date externă, formată dintr-o **mulțime ordonată de înregistrări** (elementele structurii sunt înregistrările).
- Ordonarea se face după **numărul** înregistrării care este un număr unic ce se atribuie fiecărei înregistrări atunci când este adăugată la fișier.  
**Înregistrarea curentă** este înregistrarea care se prelucrează la un moment dat.
- Fișierul este memorat pe un suport de memorie externă care se numește **volum**.
- Volumul poate conține mai multe fișiere, fiecare fișier identificându-se prin nume.


- Delimitarea fișierului pe suport se face prin **marcajele logice**:
  - **BOF** (*Begin Of File* -început de fișier) și
  - **EOF** (*End Of File* - sfârșit de fișier).
- Prelucrarea datelor se poate face în acces direct sau secvențial. Datele din fișier vor putea fi tratate ca un tot unitar sau ca date elementare independente.
- Astfel, unele operații acționează asupra întregii colecții de date (copierea, mutarea, redenumirea sau ștergerea fișierului), altele acționează asupra elementelor colecției (consultarea, actualizarea, sortarea înregistrărilor).
- În memoria externă, alocarea spațiului se face dinamic și discontinuu, în funcție de necesități, deoarece există mecanisme care, folosind informații suplimentare, permit regăsirea înregistrărilor care aparțin aceluiași fișier, indiferent de locul în care au fost scrise pe suport.

- Fișierul este o structură de date omogenă sau neomogenă (în funcție de înregistrările din care este format, care pot fi de aceeași lungime sau de lungimi diferite), externă, temporară sau permanentă, cu acces direct sau cu acces secvențial (depinde de modul în care a fost organizat pe suportul de memorare), explicită și dinamică (permite modificarea dimensiunii structurii de date după definire).
- Pentru prelucrarea datelor dintr-un fișier trebuie executate următoarele operații:
  - **deschiderea fișierului** (i se alocă o zonă de memorie internă în care sunt aduse înregistrările pentru a fi prelucrate),
  - **exploatarea fișierului** (consultarea și actualizarea structurii de date)
  - **închiderea fișierului** (eliberarea zonei de memorie alocate).
- Așadar, dacă operația acționează asupra elementelor structurii, fișierul trebuie obligatoriu deschis, iar dacă operația acționează asupra întregii structuri, fișierul trebuie să fie obligatoriu închis.


- **Înregistrarea** (*record*) este o colecție de date elementare numite câmpuri care au o legătură logică între ele (colecție de date elementare care trebuie prelucrate împreună).
- Elementele structurii se identifică prin **nume**.
- Înregistrarea este o structură de date neomogenă (elementele sale pot fi de tipuri diferite), internă sau externă (componentă a unui fișier), temporară sau permanentă, cu acces direct (este permis accesul direct la un câmp), implicită (nu este necesar să fie înregistrate informații suplimentare despre câmpuri pe suportul de memorare, care să permită regăsirea lor în cadrul structurii) și statică (structura nu mai poate fi modificată după definire).
- Asupra structurii de date de tip înregistrare se pot executa următoarele **operații**:
  - Crearea (definirea câmpurilor care o compun),
  - consultarea,
  - Actualizarea (modificarea valorii câmpurilor),
  - mutarea sau copierea în cadrul aceluiași fișier sau într-un alt fișier,
  - ștergerea din fișier.


- 
- **Câmpul** (*field*) conține un set de caractere care au legătură între ele și care formează o entitate de informație, la fel ca și o dată elementară. El păstrează aceeași categorie de informație pentru tot fișierul de date.
  - Este caracterizat de nume, tip și valoare, la fel ca orice dată elementară.
  - În unele cazuri trebuie precizată și lungimea câmpului. Numele câmpurilor pot să apară într-o expresie ca operanzi, la fel ca și numele datelor elementare, tipul operandului fiind determinat de tipul câmpului (numeric, șir de caractere, logic).


- 
- **Câmpul cheie** (*key field*) este un câmp din înregistrare care se alege pentru a identifica unic înregistrările dintr-un fișier.

- 
- **Fișierul de text** (*text file*) este fișierul în care fiecare înregistrare este formată dintr-un octet care poate conține codul unui caracter sau un cod de control:
    - returul de car - CR,
    - trecerea la linie nouă - LF,
    - tabulatorul – TAB
    - un cod de font).
  - El mai poate fi văzut ca o secvență de linii de text separate prin marcaje logice de sfârșit de linie:
    - **EOL** - End Of Line generate prin apăsarea tastei **Enter** (codurile ASCII CR+LF).
  - Prelucrarea fișierului se face secvențial. El este folosit pentru stocarea unui document care conține numai text.




- 
- **Baza de date** (*database*) este o colecție de fișiere și înregistrări legate între ele. Este mult mai avantajoasă folosirea bazelor de date în locul colecțiilor de fișiere de date.

- 
- Fiecare sistem care folosește structuri de date trebuie să pună la dispoziție comenzi, instrucțiuni sau proceduri care să permită executarea operațiilor acceptate.
  - Structurile de date de tip tablou de memorie, înregistrare și fișier de date sunt folosite de programele de aplicație. Ele sunt create în cadrul acestor programe prin intermediul limbajelor de programare.
  - Acestea pun la dispoziția programatorului instrucțiuni pentru descrierea lor (atribuirea unui nume, stabilirea componentelor).
  - Pe baza acestor informații, sistemul de operare știe să rezerve zona corespunzătoare de memorie și poate să identifice structura de date și componentele ei.
  - Limbajul de programare are implementate (sub formă de instrucțiuni, funcții sau proceduri) mecanismele prin care programatorul poate să ceară executarea diferitelor operații cu structurile de date sau cu componentele lor.

- 
- Unele produse software (de exemplu produse de calcul tabelar) *creează* și ele fișiere de date.
  - Altele, cum sunt procesoarele de texte sau editoarele de texte, creează fișiere cu texte sau fișiere document (fișierele document conțin pe lângă texte și instrucțiuni prin care se descriu formatarea documentului, formatarea paragrafelor, formatarea caracterelor, inserarea unor obiecte - imagini, sunete, diagrame etc).
  - Pentru crearea și administrarea structurilor de date de tip bază de date se folosește un instrument software specializat numit **sistem de gestiune a bazelor de date**.

- **Sistemul de gestiune a bazelor de date** pune la dispoziția utilizatorului operații specifice pentru administrarea bazelor de date: **actualizare și operații de consultare.**
- Operația de creare constă, la fel ca și la celelalte structuri de date. în definirea structurii și încărcarea ei cu date.
- Operațiile de actualizare sunt:
  - operația de inserare a unei entități în baza de date.
  - operația de ștergere a unei entități
  - operația de modificare a valorilor unei entități.
- Bazele de date relaționale sunt colecții de tabele.
- În cazul lor, operațiile de actualizare constau în:
  - inserarea unei înregistrări în tabel,
  - ștergerea unei înregistrări din tabel
  - modificarea valorilor câmpurilor din înregistrări.

- 
- Pentru a putea consulta baza de date, sistemul de gestiune a bazelor de date pune la dispoziția utilizatorului următoarele operații:
  - **Localizarea** (*locate*) înseamnă găsirea unor înregistrări în baza de date. Trebuie să fie definit un **criteriu de căutare** exprimat printr-o expresie logică ce precizează condiția pe care trebuie să o îndeplinească înregistrările căutate. După ce înregistrările au fost localizate, cu ele se pot executa diferite operații:
    - se pot afișa înregistrările sau numai anumite câmpuri din ele.
    - se aplică în toate înregistrările anumite formule de calcul asupra valorii unui câmp numeric,
    - se aplică o formulă de calcul asupra aceluiași câmp din toate înregistrările, se extrag diferite informații statistice.

- **Filtrarea** (*filter*) înseamnă extragerea din baza de date a unui grup de înregistrări. **Domeniul înregistrărilor extrase** este precizat printr-un criteriu **de filtrare** exprimat printr-o expresie logică ce precizează condiția pe care trebuie să o îndeplinească înregistrările extrase.
- **Sortarea** (*sort*) înseamnă rearanjarea înregistrărilor dintr-un tabel în ordinea crescătoare sau descrescătoare a valorilor unui câmp sau a unor câmpuri, numite **cheie de sortare**, conform unui **criteriu de sortare**. Acesta precizează cheia de sortare și ordinea de sortare.
- **Vizualizare** (*display*) înseamnă afișarea pe ecran a înregistrărilor dintr-un tabel al bazei de date.
- **Tipărirea** (*print*) înseamnă listarea la imprimantă a înregistrărilor din tabelele bazei de date. Pentru afișare se poate defini un **format de raport** (*report*) care să descrie: antetul raportului, câmpurile care vor tipărite, modul în care vor fi grupate, totalurile și subtotalurile de câmpuri numerice care se vor tipări etc.



SFÂRȘIT