

1. Introducere în BD

- anii 60-70 - începuturi

- BD propriu-zisă = colecție de inform. stocate

- programe de aplicație = progr. care manipulează BD

- BD li. să fie persistentă (să fie stocate tot timpul) ⇒ să fie stocată pe memorie externă

- TABELĂ = este o colecție de date structurată, org. în articole, care se potrivește similar cu tabelele stocate pe format de hârtie (tabele manuale). Tabela este principala entitate din structura unei BD.

- pt. tabelele manuale avem:

a) capul de tabel

b) rânduri (înregistrări)

- pt. tabele: capul de tabel s.u. structura tabelă, def. prin 3 componente:

a) nume coloană (câmp = field)

b) tip dată (def. pt. fiecare coloană)

c) format (asociat tipului de dată)

OBS: formatul se referă la formatul de afișare nu la modalitatea de stocare internă.

• nume coloană (câmp) = se definește similar cu modul de def al unei variabile într-un li. de progr., e format dintr-un șir de caractere cu restricții legate de caracterele de început, dependente de mediul de BD. Numele are un nr. de caractere ce depinde de mediul de BD (ex: max 10, max 20 sau 32 de caractere). Se recomandă ca numele unei coloane să fie sugestiv - să indice semnificația inform. stocate în coloana respectivă.

• Tipuri de date: sunt utilizate ≠ tipuri de date funcție de inform. stocate.

① CHAR (STRING) - în unele medii de BD este CHAR, în altele STRING; pt. păstrarea unui caracter se declară de tipul CHAR (1);

{ CHAR(20) = stringul format din ^{max} 20 de caractere;

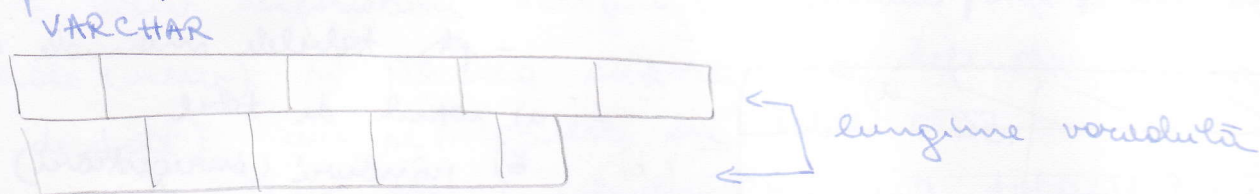
{ CHAR(1) = un caracter

STRING(20) = șirul format din 20 de caractere;
- în Oracle, Microsoft SQL - CHAR;

CHAR(20) - interpretare: pt. coloana respectivă s-a rezervat spațiu pt 20 de caractere. În coloana respectivă, toate înregistrările au lungime fixă, la stocarea tabelului se ocupă spațiul din unele înregistrări nu au 20 de caractere \Rightarrow BD non-densă.

VARCHAR(n) - indică lungimea variabilă în care se prezintă doar limita superioară (n).

ex: VARCHAR(25) = câmpul are tipul de date - șir de caractere, limitat superior la 25, dar de. inform. are mai puțin de 25 caract. va fi compactată.



Avantaje / Dezavantaje CHAR / VARCHAR:

- înregistrări de lung fixă (CHAR) \leftarrow dezavantaj: sp. de stocare mare
avantaj: viteza de acces f. mare
- VARCHAR \leftarrow avantaj: spațiu de stocare optimizat
dezavantaj: acces dificil

- limitarea superioară $n=25$: în tabelă inform. e compactată, însă în structura de index se ține cont de limite maximă (pt a se face accesul la o anumită înregistrare).

OBS: NU se face distincție între caractere și șirul de caractere.

② NUMERIC

- INTEGER: tip de date întreg repres. pe 16 biți în ~~mem.~~ semn;
- NATURAL: 16 biți, dar nu suportă semnul
- NUMBER(m,n) - uzual

$\left\{ \begin{array}{l} m = \text{numărul nr. total de digits (inclusiv punct zecimal)} \\ n = \text{nr. digits după punctul zecimal} \end{array} \right.$

(7.2) \rightarrow $\underbrace{x \ x \ x \ x}_{7} . \overbrace{x \ x}^2$

- FLOAT - virgulă mobilă

- format științific: $2.92 \ E+27$ ($2.92 \cdot 10^{27}$).

- CURRENCY (similar cu NUMBER) - se poate defini un simbol implementat asociat la valoare (2,3 \$)

DATE - dată calendaristică, are în general format simplu;
formatul simplu (intern) are 10 caractere:

{ m m / d d / y y y y }

{ m = month
d = day
y = year
/ = separator

- formatul de afişare pt DATE
este selectabil;

④ TIME - nu se registrează în toate mediile BD; formatul este:

h h : m m : s s . x . . . x

format simplu

nr. de cifre: variabil pt. zecimile de secundă; sunt generate şi păstrate numai intern

- zecimile de secunde sunt păstrate pt. jurnalizarea (detaliată) a
bazei de date; în cazul secunzii conuenit la BD.

⑤ BOOLEAN (LOGICAL) = pp. păstrarea a 2 valori < T = true (1)

- în toate mediile - este ocupat 1 octet pt. a tine valoarea în boolean
(deci se fi suficient 1 bit).
F = false (0)

⑥ MEMO - se substitueşte adesea cu VARCHAR.

TABELĂ:

			OBS.
			*1
			*2

se lungime ≠ sau duce minime

*1 = 500 caractere

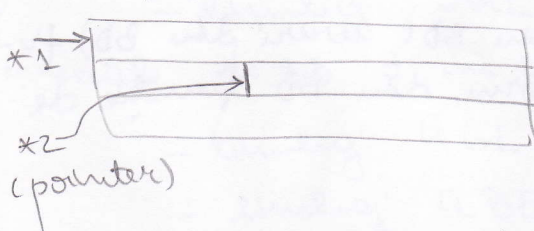
*2 = 20 caractere

la restul - nu am minime

- tabelul NU suportă VARCHAR mai de 10, să
rezerv 500 de caractere

- OBS a fost declarat de tip MEMO

- fişierul extern de tip text (Maxtext, unde după altul) tine
minime inform care este ≠ de null în coloane respectivă.



- numele fişierului este dat intern
(cu un stiv de minime etc);

Formatul pt MEMO e dependent de dom, spaţiului de stocare,
pointerul este dimensionat la 8 digiti

- pe baza fişierului pointer asociat cu câmpul OBS se merge
în fişierul text şi se aduce informaţie.

OBS: NU se poate face o căutare după informaţie propriu-zisă.

⑦ BLOB (L BLOB) - are din nou în suportă lung
(long blob) dimensiuni, L 4MB, 8MB, 16MB

ex: adăugarea pt o persoană - se stochează în BLOB.

Inform. păstrată în rândurile tabelului s.u. record = înregistrare sau row (rând).

OBS: nu în medie de BD în suportă se adăugă să fie segmentată pe 2 rânduri.

ex: într-un char de 20 NU sau ENTER (ENTER th. convertit la BLANK).

Indicator de înregistrare: indică recordul curent. Poate avea valoarea
TOP = prima înregistrare fizică din tabelă
BOTTOM
n - slot între TOP și BOTTOM

ex: n=3 \Rightarrow indicat de înreg. indică înregistrarea a3-a.

OBS: A restat în cu a paranteze nr. de înregistrări dintr-o tabelă (APPEND \neq INSERT). de. nr. de înregistrări vechi, performanțele BD scad.

Bază de date = colecție de tabele între care există relații sau asocieri urmărind reducerea redundanței pt. stocarea datelor și creșterea performanțelor din punctul de vedere al accesului la date.

- redundanță \neq mare: BD conține o singură tabelă

- redundanță măsurată: BD conține nr. mare de tabele (fără cu câte 2 câmpuri); accesul devine dificil.

- se urmărește scăderea redundanței (să nu păstrăm același lucru de mai multe ori în recorduri \neq) și creșterea vitezei de prelucrare; tendințele sunt contradictorii și th. găsit un optim.

Mediu de BD = pachet de programe ce permite definirea BD (tabele, structuri, relații), manipularea inform. din BD (cărți din BD) precum și utilitățile ce permit dezvoltarea inform. din BD funcție de cerințele definite.

Mediul permite:

- crearea BD

- scrierea programelor aplicative - pentru munca cu unele BD.

+ mediu de bază de date sau inclus SGBD sau DBMS = Data Base Management System.

existențe medii BD:

① După cost (de licențiere):

- medii profesionale (> 3000\$)

ORACLE, PROGRESS, DB2, ... - permit manipularea BD de dimensiuni mari, cu facilități de securizare, multi-user etc., dimensiuni de TB;

• DB2 - prima bază de date dezvoltată de IBM

• ORACLE = mediul de BD al firmei cu același nume, suportă o multitudine de platforme

• PROGRESS - găsit în 1988 și a fost dezvoltat de inginerii ai lui ORACLE, a-a doborât rezoluția problemelor de servere de aplicații pe care le avea ORACLE DB2 și ORACLE - sunt motive SQL, dar PROGRESS - 4GL (limbaj de manipulare).

- ERP - BD și informații (Enterprise Resource Planning)

- Informații bancare

ERP ← SAP = aplicație din Germania (ORACLE) - parte Europa

MFG PRO (PROGRESS) - SUA

- în cazul inform. bancare - securitatea este f. importantă; cu costuri f. mari.

- medii comerciale - aplicații de mici dimensiuni, cu costuri relativ mici
ACCESS, DBASE, FOX, Microsoft SQL, ...

- medii open source (free)

MySQL, ...

↳ preferat pt. aplicații WEB, cu bază de date

② După limbajul de manipulare a datelor (DML = Data Manipulation Language):

- limbaj SQL (Structured Query Language) - ~95% din mediile de BD în prezent;

- limbaj 4GL (4 Generation Language)

- limbaj ABE (Query By Example) - pe ideea de duplicare pt. ca a fost destinat începătorilor.

③ După limbajul dezvoltare aplicații:

- limbaj motor: oferă facilități de servere BD dar și dezvoltarea interfețelor, aplicații;

- limbaj gazdă: Gupta, Raima. - nu oferă facilități la interfețe și aplicații

④ După modelul aplicației:

- medii client-server
- medii non-client-server.

⑤ După numărul utilizatori simultani la BD:

- medii mono-user
- medii multi-user.

⑥ După interfața utilizator:

- interfața alfa-numericală (pt. profesioniști).
- interfața grafică - nu sunt avantaje dpdv al BD (pt utilizatorii mei)

⑦ După platformă:

- medii pe platformă Windows (Access, MySQL)
- LINUX
- UNIX
- mediile de BD puternice (ORACLE) sunt multipatformă

Caracteristici suplimentare pt. medii de BD:

a) rollback / rollforward - facultatea de a se BD anula sau de a o aduce la zi; după un rollback, se face rollforward care înlocuiește modificările aduse BD

b) medii tranzacționale: facultatea de a defini un șir de operații ca o tranzacție.

(- operațiile sunt ținute în buffere \Rightarrow COMMIT \Rightarrow sunt în bază)

c) securitate internă: mediul de BD are propriul nivel de securitate a datelor; drepturi diferite pt. utilizatori pt. comenzi record (unele sunt imposibile, altele pot face doar read, doar write, nu permite ștergere, nu permite append etc.).

d) integritate consistentă: posibilitatea de a defini constrângeri pt. tabele și care să fie verificate de SGBD.

- ... în BD (cuprinde tabelele BD):
- READ: ne vizualizează conținutul unei tabele / înregistrărilor
 - FIND: găsește înregistrări care îndeplinesc un anumit criteriu, FIND se referă la găsirea primei înreg. care îndepl. criteriul.
 - FIND NEXT = găsește urm. înreg. care îndepl. crit.
 - WRITE (MODIFY, UPDATE) = pot să modifice conținutul unei înregistrări existente;
 - APPEND } adăugarea unei înregistrări noi
 - INSERT }
 - APPEND sugerează la sf. tabelului și este mai des utilizat, INSERT pp la reorganizare tabelă;
 - SORT - recomandată, durează f. mult, organizează tabele după un criteriu → ordonare fizică;
 - INDEX - nu necesită ordonare fizică, ci realizează o ordonare logică (permite un criteriu de acces logic la tabele fizice).
 - DELETE - ștergere înregistrare / înregistrări.
 - modificare structură: adăugare coloană, ștergere coloană, este în general o operație nepermisă, fiind executor de SGBD.
 - bazele view - tabele virtuale, păstrate ca definiție, nu ca date.

2. Date Base Management System - DBMS = SGBD

DBMS = componenta centrală a unui MBD (mediu de lucru de date), Rolul DBMS este de a furniza instrumente pentru crearea bazei de date, crearea tabelelor într-o BD, definirea structurii datelor din tabelă, instrumente legate de manipularea datelor din tabelă (implementarea limbajului de cereri), instrumente legate de modul de stocare a datelor pe memorie externă, instrumente utilizate pt. definirea și manipularea view-ului, instrumente legate de administrarea BD.

Instrumentele sunt grupate, după funcțiile îndeplinite, astfel:

- DDL = Data Definition Language: gestionează struct. specifi. a DBMS-ului ~~unde~~ se permite definirea BD
- DML = Data Manipulation Language: interpretează cererile de manipulare la bază.

OBS: DML este standardizat, limbajul de cereri fiind același pt. toate mediile. Performanțele asupra manipulării unei BD sunt dictate de DML.

- SDL = Storage Definition Language: gestionează modul de stocare al datelor pe memoria externă. Nu se regăsește în toate RDB.
- VDL = View Definition Language: permite definirea tabelor virtuale (view) și stocarea surselor (se definește, nu se stocază de date). Nu se regăsește în toate RDB.
- DBA = Data Base Administrator: reuneste totalitatea instrumentelor privind definirea și controlul accesului la date.

DDL + DML + SDL + VDL = limbaj SQL.

OBS: procedurile SDL sunt în general rare și utilizate numai de DBA. SDL este f. important la volume f. mari de date.

DBMS = DBMS - managementul datelor