



Algebra relațională

- 
- Reuniune, intersecție, diferență
 - Redenumire, selecție, proiecție, joncțiune
 - Interogări în algebra relațională
 - Valori NULL în algebra relațională
 - Vederi

- 
- **Actualizare** - funcție care pornind de la o bază de date produce o altă bază de date (fără a-i modifica schema).
 - **Interogare** - funcție care returnează o relație, plecând de la o bază de date.



Algebra relațională

- Se bazează pe o colecție de operatori ce sunt aplicații relațiilor, producând relații.

Operatorii algebrici relaționali se împart în următoarele clase:

- Operatori clasici din teoria mulțimilor – reuniune, intersecție, diferență;
- Operatori de redenumire, selecție, proiecție;
- Operatorul join (joncțiune), împreună cu variantele sale – *joncțiune naturală, produs cartezian, theta joncțiune și joncțiune externă.*

Reuniune, intersecție, diferență - se aplică doar asupra perechilor de relații definite pe aceleași attribute.

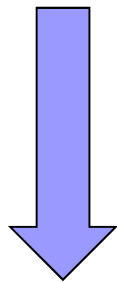
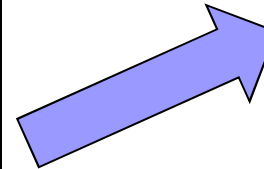
GPS

Producător	MODEL
Nokia	N2310
Samsung	S5100
Samsung	S5600
Nokia	N4215

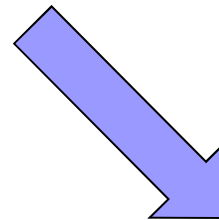
RADIO

Producător	MODEL
Nokia	N4215
Samsung	S2110
Samsung	S5600
Samsung	S1100

Reuniune?



Intersecție?



Diferență?

Definiții

Reuniunea a două relații $r_1(X)$ și $r_2(X)$ este relația notată $r_1 \cup r_2$ care conține tupluri ce aparțin fie lui r_1 fie lui r_2 , fie ambelor relații r_1 și r_2 .

Intersecția dintre relațiile $r_1(X)$ și $r_2(X)$ este relația notată $r_1 \cap r_2$ formată din tuplurile comune relațiilor r_1 și r_2 .

Diferența dintre relațiile $r_1(X)$ și $r_2(X)$ este relația notată $r_1 - r_2$ care conține tuplurile din r_1 care nu se regăsesc în r_2 .

TATA_COPIIL

Tata	Copil
Adam	Dan
Adam	Marian
Radu	Cristi
Radu	Catalin

MAMA_COPIIL

Mama	Copil
Eva	Dan
Eva	Lucian
Maria	Cristi
Carmen	Catalin

TATA_COPIIL \cup **MAMA_COPIIL** ??

Operația de redenumire (ρ)

Limitările impuse operatorilor standard din teoria mulțimilor pot fi restrictive în anumite situații.

Exemplu

- Să considerăm cele două relații din figura 3.1.

TATA_COPIIL

Tata	Copil
Adam	Dan
Adam	Marian
Radu	Cristi
Radu	Catalin

MAMA_COPIIL

Mama	Copil
Eva	Dan
Eva	Lucian
Maria	Cristi
Carmen	Catalin

TATA_COPIIL \cup **MAMA_COPIIL** ??

Figura 3.1 Reuniune logică, dar incorectă

Ar avea sens executarea unei reuniuni între cele două relații cu scopul de a obține toate perechile 'părinte – copil' din baza de date.

Acest lucru nu este însă posibil deoarece atributul denumit de noi 'părinte' are numele 'tata' într-o relație și numele 'mama' în cealaltă relație.

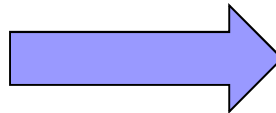
Operatorului de redenumire – adaptează numele atributelor atunci când este necesară aplicarea unui operator din teoria mulțimilor.

Obs. Acest operator modifică numele atributelor lăsând intact conținutul relației.

Exemplu

TATA_COPII

Tata	Copil
Adam	Dan
Adam	Marian
Radu	Cristi
Radu	Catalin



$\rho_{Parinte \leftarrow Tata}$ (**TATA_COPII**)

Parinte	Copil
Adam	Dan
Adam	Marian
Radu	Cristi
Radu	Catalin

Fig.3.2 Exemplu de redenumire

Definiție

Fie r o relație definită pe mulțimea X de atribute și fie Y o altă mulțime de atribute având aceeași cardinalitate ca și X ; fie secvențele ordonate și $A_1A_2\dots A_k$, $B_1B_2\dots B_k$ formate din atributele mulțimii X respectiv Y .

Redenumirea $\rho_{A_1A_2\dots A_k \leftarrow B_1B_2\dots B_k}(r)$ conține pentru fiecare tuplu t din r un tuplu t' definit pe Y astfel încât $t'[B_i] = t[A_i]$, $i = \overline{1, k}$.

În practică, cele două secvențe $A_1A_2\dots A_k$ și $B_1B_2\dots B_k$ vor conține doar atributele ce urmează a fi redenumite și noile lor nume.

Operatorul de selecție (σ)

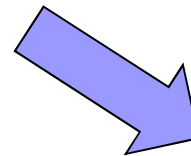
Rezultatul unei selecții conține tuplurile relației operand ce satisfac condiția asociată operatorului.

Exemplu

Se consideră relația ANGAJAT din figura următoare:

ANGAJAT

Nume	Prenume	Varsta	Salariu
Ionescu	Maria	25	2000
Popescu	Lucia	40	3000
Diaconescu	Nicu	36	4500
Ionescu	Marin	40	3900



$\sigma_{Varsta < 30 \vee Salariu > 4000}$ (ANGAJAT) ??

Operatorul de selecție (σ)

Exemplu

ANGAJAT

Nume	Prenume	Varsta	Salariu
Ionescu	Maria	25	2000
Popescu	Lucia	40	3000
Diaconescu	Nicu	36	4500
Ionescu	Marin	40	3900



$\sigma_{Varsta < 30 \vee Salariu > 4000}$ (**ANGAJAT**)

Nume	Prenume	Varsta	Salariu
Ionescu	Maria	25	2000
Diaconescu	Nicu	36	4500

Fig.3.3 Exemplu de selecție

După cum se poate observa, sunt selectate tuplurile din relația ANGAJAT care îndeplinesc condiția: **Varsta < 30** sau **Salariu > 4000** (formulă propozițională).

Definiții

Fiind dată o relație r definită pe o mulțime de atribute X , spunem că o formulă propozițională F definită pe X este o formulă compusă din condiții atomice de tipul $A\theta B$ sau $A\theta c$ legate prin conectorii \vee (OR), \wedge (AND) și \neg (NOT), în care:

- θ este un operator de comparație ($=, \neq, <, \leq, >, \geq$);
- A și B sunt atribute din X compatibile (conțin valori pentru care are sens comparația);
- c este o constantă compatibilă cu domeniul atributului A .

Definiții

Fiind dată formula propozițională F și tuplul t , se definește *valoarea de adevăr a lui F pe t* :

- $A\theta B$ este adevărată dacă și numai dacă $t[A]$ este în relația θ cu $t[B]$;
- $A\theta c$ este adevărată dacă și numai dacă $t[A]$ este în relația θ cu c ;
- $F_1 \vee F_2$, $F_1 \wedge F_2$ și $\neg F_1$ au semnificația uzuală.

O selecție $\sigma_F(r)$ produce o relație care are aceleași atribute ca relația r și care conține acele tupluri din r pentru care F este adevărată.

Operatorul de proiecție (π)

Definiție

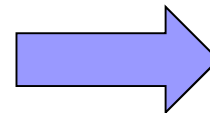
Fie dată relația r definită pe mulțimea de atribute X și o submulțime Y a lui X .

Proiecția relației r pe Y (notată $\pi_Y(r)$) reprezintă mulțimea tuplurilor definite pe Y obținute din tuplurile lui r luând în considerare numai valorile corespunzătoare atributelor din Y :

$$\pi_Y(r) = \{t[Y] / t \in r\}$$

ANGAJAT

Nume	Prenume	Departament	Sef
Ionescu	Maria	Vanzari	Luca
Popescu	Lucia	Vanzari	Luca
Diaconescu	Maria	Personal	Damian
Ionescu	Marin	Personal	Damian



$\pi_{\text{Departament, Sef}}$ (ANGAJAT) ??

Proiecție

Situații în care numărul de tupluri ale rezultatului este egal, respectiv mai mic decât numărul de tupluri ale operandului:

ANGAJAT

Nume	Prenume	Departament	Sef
Ionescu	Maria	Vanzari	Luca
Popescu	Lucia	Vanzari	Luca
Diaconescu	Maria	Personal	Damian
Ionescu	Marin	Personal	Damian

$\pi_{\text{Departament, Sef}}$ (ANGAJAT)

Departament	Sef
Vanzari	Luca
Personal	Damian

Fig.3.4 Exemplu de proiecție cu mai puține tupluri decât operandul

ANGAJAT

Nume	Prenume	Varsta	Salariu
Ionescu	Maria	25	2000
Popescu	Lucia	40	3000
Diaconescu	Nicu	36	4500
Ionescu	Marin	45	3900

$\pi_{\text{Varsta, Salariu}}$ (ANGAJAT)

Varsta	Salariu
25	2000
40	3000
36	4500
45	3900

Fig.3.5 Exemplu de proiecție cu același număr de tupluri ca și operandul

Întrebare:

Când $\pi_Y(r)$ conține același număr de tupluri ca și r ?

Propoziție

$\pi_Y(r)$ conține același număr de tupluri ca și r dacă și numai dacă Y este o super-cheie pentru r .

Operatorul joncțiune – „join”

- Permite realizarea unei conexiuni între datele conținute de diverse relații.

Joncțiunea naturală () – „natural join” - corelează datele din relații diferite pe baza **valorilor egale asociate atributelor cu același nume.**

CONTRAVENTIE

Cod	Data	Cadru	Judet	NrInmat
143256	25/10/92	567	IS	02 AAA
987554	26/10/92	456	IS	02 AAA
987557	26/10/92	456	IS	03 BBB
630876	15/10/92	456	VS	03 BBB
539856	12/10/92	567	VS	03 BBB

AUTOVEHICUL

NrInmat	Judet	Proprietar	Adresa
03 BBB	IS	Maftai Eduard	Nicolina 30
01 CCC	IS	Maftai Eduard	Nicolina 30
02 AAA	IS	Luca Marian	Primaverii 4
03 BBB	VS	Melinte Dan	Primaverii 17

CONTRAVENTIE ⋈ AUTOVEHICUL

Cod	Data	Cadru	Judet	NrInmat	Proprietar	Adresa
143256	25/10/92	567	IS	02 AAA	Luca Marian	Primaverii 4
987554	26/10/92	456	IS	02 AAA	Luca Marian	Primaverii 4
987557	26/10/92	456	IS	03 BBB	Maftai Eduard	Nicolina 30
630876	15/10/92	456	VS	03 BBB	Melinte Dan	Primaverii 17
539856	12/10/92	567	VS	03 BBB	Melinte Dan	Primaverii 17

Operatorul joncțiune – „join”

- Permite realizarea unei conexiuni între datele conținute de diverse relații.
- Există două versiuni principale ale acestui operator, care, oricum, se pot obține una din cealaltă.

Joncțiunea naturală (\bowtie) – „natural join” - corelează datele din relații diferite pe baza valorilor egale asociate atributelor cu același nume.

Definiție

Fie $r_1(X_1)$ și $r_2(X_2)$ două relații. *Joncțiunea naturală* $r_1 \bowtie r_2$ este o relație definită pe X_1X_2 (reuniunea dintre X_1 și X_2) astfel încât:

$$r_1 \bowtie r_2 = \{t \text{ definit pe } X_1X_2 / \exists t_1 \in r_1, \exists t_2 \in r_2, \text{ a. i. } t[X_1] = t_1 \text{ si } t[X_2] = t_2\}$$

Pe scurt putem scrie:

$$r_1 \bowtie r_2 = \{t \text{ definit pe } X_1X_2 / t[X_1] \in r_1 \text{ si } t[X_2] \in r_2\}$$

Exemplu

CONTRAVENTIE

Cod	Data	Cadru	Judet	NrInmat
143256	25/10/92	567	IS	02 AAA
987554	26/10/92	456	IS	02 AAA
987557	26/10/92	456	IS	03 BBB
630876	15/10/92	456	VS	03 BBB
539856	12/10/92	567	VS	03 BBB

AUTOVEHICUL

NrInmat	Judet	Proprietar	Adresa
03 BBB	IS	Maftai Eduard	Nicolina 30
01 CCC	IS	Maftai Eduard	Nicolina 30
02 AAA	IS	Luca Marian	Primaverii 4
03 BBB	VS	Melinte Dan	Primaverii 17

CONTRAVENTIE \bowtie AUTOVEHICUL

Cod	Data	Cadru	Judet	NrInmat	Proprietar	Adresa
143256	25/10/92	567	IS	02 AAA	Luca Marian	Primaverii 4
987554	26/10/92	456	IS	02 AAA	Luca Marian	Primaverii 4
987557	26/10/92	456	IS	03 BBB	Maftai Eduard	Nicolina 30
630876	15/10/92	456	VS	03 BBB	Melinte Dan	Primaverii 17
539856	12/10/92	567	VS	03 BBB	Melinte Dan	Primaverii 17

Joncțiunea naturală a celor două relații s-a obținut prin combinarea fiecărui tuplu din CONTRAVENTIE cu exact un tuplu din AUTOVEHICUL:

- cu cel mult unul deoarece attributele Judet și NrInmat formează o cheie a relației AUTOVEHICUL;
- cu cel puțin unul datorită constrângerii de referință dintre attributele Judet și NrInmat din CONTRAVENTII și relația AUTOVEHICUL.

Definiție. Fie $r_1(X_1)$ și $r_2(X_2)$ două relații. Spunem că **joncțiunea naturală** $r_1 \bowtie r_2$ este completă dacă:

$$\forall t_1 \in r_1, \exists t \in r_1 \bowtie r_2, \text{ a.i. } t[X_1] = t_1$$

$$\forall t_2 \in r_2, \exists t' \in r_1 \bowtie r_2, \text{ a.i. } t'[X_2] = t_2$$

r_1

Angajat	Departament
Ionescu	vanzari
Balint	productie
Baltag	productie

r_2

Departament	Sef
productie	Manole
vanzari	Burlacu

$r_1 \bowtie r_2$

Angajat	Departament	Sef
Ionescu	vanzari	Burlacu
Balint	productie	Manole
Baltag	productie	Manole

Fig.3.6 Exemplu de joncțiune naturală completă

r_1

Angajat	Departament
Ionescu	vanzari
Balint	productie
Baltag	productie

 r_2

Departament	Sef
productie	Manole
depanare	Burlacu

 $r_1 \bowtie r_2$

Angajat	Departament	Sef
Balint	productie	Manole
Baltag	productie	Manole

Exemplu de joncțiune naturală incompletă

r_1

Angajat	Departament
Ionescu	vanzari
Balint	productie
Baltag	productie

 r_2

Departament	Sef
marketing	Manole
depanare	Burlacu

 $r_1 \bowtie r_2$

Angajat	Departament	Sef

Exemplu de joncțiune vidă

Joncțiunea externă – „outer join”

Operatorul de joncțiune exclude tuplurile unei relații ce nu au corespondent în celălalt operand.

Operatorul de joncțiune externă - asigură prezența în relația rezultat a tuturor tuplurilor unei relații, acestea fiind completate cu valori NULL atunci când nu au corespondent în cealaltă relație.

Acest operator are trei variante :

- joncțiune externă la stânga – extinde doar tuplurile primului operand;
- joncțiune externă la dreapta – extinde doar tuplurile celui de-al doilea operand;
- joncțiune externă completă – extinde toate tuplurile.

Exemplu

 r_1

Angajat	Departament
Ionescu	vanzari
Balint	productie
Luca	productie

 r_2

Departament	Sef
productie	Manole
achizitii	Burlacu

 $r_1 \bowtie_{\text{LEFT}} r_2$

Angajat	Departament	Sef
Ionescu	vanzari	null
Balint	productie	Manole
Luca	productie	Manole

 $r_1 \bowtie_{\text{RIGHT}} r_2$

Angajat	Departament	Sef
Balint	productie	Manole
Luca	productie	Manole
null	achizitii	Burlacu

 $r_1 \bowtie_{\text{FULL}} r_2$

Angajat	Departament	Sef
Ionescu	vanzari	null
Balint	productie	Manole
Luca	productie	Manole
null	achizitii	Burlacu

Fig.3.7 Exemple de joncțiuni externe

Theta-joncțiune și echi-joncțiune

În general produsul cartezian nu prezintă interes deoarece combină tuplurile celor doi operanzi într-o manieră lipsită de semnificație.

ANGAJAT

Angajat	Proiect
Ionescu	A
Balint	A
Balint	B

PROIECT

Cod	Nume
A	Venus
B	Marte

ANGAJAT ⋈ PROIECT

Angajat	Proiect	Cod	Nume
Ionescu	A	A	Venus
Balint	A	A	Venus
Balint	B	A	Venus
Ionescu	A	B	Marte
Balint	A	B	Marte
Balint	B	B	Marte

Figura 3.8 Produs cartezian

Operatorul de **theta-joncțiune** - produs cartezian urmat de o selecție.

$$r_1 \bowtie_F r_2 = \sigma_F(r_1 \bowtie r_2)$$

F este condiția de selecție

ANGAJAT

Angajat	Proiect
Ionescu	A
Balint	A
Balint	B

PROIECT

Cod	Nume
A	Venus
B	Marte

$\sigma_{Proiect=Cod}$ (ANGAJAT \bowtie PROIECT)

Angajat	Proiect	Cod	Nume
Ionescu	A	A	Venus
Balint	A	A	Venus
Balint	B	B	Marte

Figura 3.9 Produs cartezian urmat de o selecție

Echi-joncțiunea - o theta-joncțiune în care condiția de selecție F este o conjuncție de atomi de egalitate, fiecare atom implicând câte un atribut din fiecare operand.

- A treia relație din figura anterioară - rezultat al unei echi-joncțiuni.

Obs.

1. Majoritatea sistemelor de gestiune a bazelor de date nu exploatează avantajele numelor atributelor în combinarea relațiilor \Rightarrow theta și echi-joncțiunea au o importanță deosebită.
2. interogările SQL corespund echi – joncțiunilor.
3. Joncțiunea naturală a devenit disponibilă doar în versiunile recente de SQL.



Interogări în algebra relațională

- **Interogarea** - funcție care, aplicată asupra unei instanțe a unei baze de date, produce o relație;
 - Fiind dată o schemă R a unei baze de date, o interogare este o funcție care, pentru fiecare instanță r a lui R , produce o relație definită pe o mulțime X de attribute.
- În algebra relațională, interogările unei scheme R de baze de date sunt formulate cu ajutorul unor expresii, ale căror atomi sunt relații din R .

Exemple

Se consideră o bază de date formată din relațiile:

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

ANGAJAT

NrInreg	Nume	Varsta	Salariu
101	Maria Ionescu	34	40
103	Maria Balint	23	35
104	Lucia Popescu	38	61
105	Nicu Luca	44	38
210	Marcel Burlacu	49	60
231	Alin Lupu	50	60
252	Nicu Luca	44	70
301	Andrei Popa	34	70
375	Maria Ionescu	50	65

SUPERVIZOR

NrSup	NrAng
210	101
210	103
210	104
231	105
301	210
301	231
375	252

Fig.3.10 Bază de date pentru exemplificarea interogărilor în algebra relațională

1) Să se găsească numerele de înregistrare pentru supervizorii acelor angajați ce câștigă mai mult de 40.

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

$$\pi_{NrSup} (SUPERVIZOR \bowtie_{NrAng=NrInreg} (\sigma_{Salariu>40} (ANGAJAT)))$$

2) Să se găsească numerele de înregistrare și numele supervizorilor ce au numai subalterni ce câștigă mai mult de 40.

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

- se găsesc toți supervizorii pentru care există subalterni care câștigă maxim 40;

$$\pi_{NrSup} (SUPERVIZOR \bowtie_{NrAng=NrInreg} (\sigma_{Salariu \leq 40} (ANGAJAT)))$$

ANGAJAT

NrInreg	Nume	Varsta	Salariu
101	Maria Ionescu	34	40
103	Maria Balint	23	35
104	Lucia Popescu	38	61
105	Nicu Luca	44	38
210	Marcel Burlacu	49	60
231	Alin Lupu	50	60
252	Nicu Luca	44	70
301	Andrei Popa	34	70
375	Maria Ionescu	50	65

SUPERVIZOR

NrSup	NrAng
210	101
210	103
210	104
231	105
301	210
301	231
375	252

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

- se aplică operatorul de diferență între submulțimea supervizorilor și mulțimea obținută la pasul anterior.

$$\pi_{NrInreg, Nume} (ANGAJAT \bowtie_{NrInreg=NrSup} (\pi_{NrSup} (SUPERVIZOR) -$$

$$\pi_{NrSup} (SUPERVIZOR \bowtie_{NrAng=NrInreg} (\sigma_{Salariu \leq 40} (ANGAJAT))))))$$

ANGAJAT

NrInreg	Nume	Varsta	Salariu
101	Maria Ionescu	34	40
103	Maria Balint	23	35
104	Lucia Popescu	38	61
105	Nicu Luca	44	38
210	Marcel Burlacu	49	60
231	Alin Lupu	50	60
252	Nicu Luca	44	70
301	Andrei Popa	34	70
375	Maria Ionescu	50	65

SUPERVIZOR

NrSup	NrAng
210	101
210	103
210	104
231	105
301	210
301	231
375	252

Valori NULL în algebra relațională

În cele discutate anterior am presupus că expresiile algebrice sunt aplicate unor relații ce nu conțin valori NULL.

Ținând cont de importanța valorilor NULL în aplicații, vom vedea care este impactul lor asupra limbajelor de interogare și actualizare a datelor.

Exemplu. Să considerăm relația din figura următoare

PERSOANA

Nume	Varsta	Salariu
Ionescu	35	500
Popescu	27	600
Popa	NULL	500

Figura 3.12 Relație cu valori NULL

și selecția $\sigma_{Varsta > 30}(PERSOANA)$

- Primul tuplu va contribui la rezultatul selecției, iar al doilea tuplu nu.
- Ce putem spune despre al treilea tuplu (valoarea NULL având semnificația unei informații pe care o ignorăm)?

Valori NULL în algebra relațională

În raport cu interogările prezentate anterior vom utiliza o logică **trivalentă** în locul celei bivalente:

- O formulă poate avea valorile de adevăr **TRUE (T)**, **FALSE (F)** sau **UNKNOWN (U)**.
- Rezultatul unei condiții atomice va avea valoarea de adevăr **UNKNOWN** dacă cel puțin un termen al comparației are valoarea **NULL**.

Revenind la exemplul anterior, expresia va produce o relație formată din primul tuplu, pentru care valoarea de adevăr a expresiei a fost **TRUE**.

Tabelele de adevăr în logica trivalentă pentru conectorii **not**, **and** și **or** sunt:

<i>not</i>	
U	U
F	T
T	F

<i>and</i>	T	U	F
T	T	U	F
U	U	U	F
F	F	F	F

<i>or</i>	T	U	F
T	T	T	T
U	T	U	U
F	T	U	F

Valori NULL în algebra relațională

Este de notat că această logică trivalentă pentru operatorii algebrici prezintă unele dezavantaje.

Exemplu

Să considerăm expresia

PERSOANA

Nume	Varsta	Salariu
Ionescu	35	500
Popescu	27	600
Popa	NULL	500

$$\sigma_{Varsta>30}(PERSOANA) \cup \sigma_{Varsta\leq 30}(PERSOANA)$$

- Logic, această expresie ar trebui să returneze relația PERSOANA.
- Pe de altă parte, dacă cele două subexpresii sunt evaluate separat, al treilea tuplu va produce un rezultat necunoscut pentru fiecare subexpresie, deci și pentru reuniune.
- Numai prin intermediul unei evaluări globale (abordare care nu este practică în cazul expresiilor complexe) putem ajunge la concluzia că un astfel de tuplu trebuie să apară în rezultat.

Valori NULL în algebra relațională

Cea mai bună metodă practică de a depăși astfel de dificultăți este de a trata valorile NULL din punct de vedere pur sintactic.

În acest sens sunt introduse două noi condiții atomice de selecție pentru a verifica dacă o valoare este specificată sau este NULL:

- **A is NULL** este adevărată pentru tuplul t dacă $t[A] = NULL$ și falsă în rest;
- **A is NOT NULL** este adevărată pentru tuplul t dacă $t[A] \neq NULL$ și falsă în rest.

Exemplu

$\sigma_{Varsta > 30}(PERSONA)$ - returnează persoanele cu vârsta peste 30 ani.

$\sigma_{Varsta > 30 \vee Varsta \text{ IS NULL}}(PERSONA)$ - returnează persoanele care au sau care pot avea peste 30 ani.

Această abordare este utilizată în versiunile recente de SQL care suportă logica trivalentă.



VEDERI

În cele prezentate anterior am văzut că se pot construi reprezentări diferite ale datelor, reprezentări ce vor fi disponibile utilizatorilor prin intermediul *relațiilor derivate*.

Tipuri de relații într-o bază de date relațională

- ❑ **relații de bază** - conținutul este autonom și stocat în baza de date
- ❑ **relații derivate** - conținutul este definit pe baza conținutului altor relații

Există două tipuri principale de relații derivate:

- **vederi materiale** - relații derivate stocate în baza de date;
- **vederi (relații virtuale)** – relații definite prin intermediul unor funcții (expresii ale limbajului de interogare) care nu sunt stocate în baza de date, dar pot fi folosite în interogări ca și cum ar exista fizic.

Vederile materiale oferă un avantaj când numărul cererilor de interogare este mai mare decât operațiile de actualizare ale relației pe care se bazează vederea.

Deoarece nu se pot specifica tehnici generale de păstrare a consistenței între relațiile de bază și vederile materiale, **majoritatea sistemelor comerciale oferă mecanisme numai pentru relațiile virtuale (vederi).**

Vederile sunt definite în sistemele relaționale ca fiind expresii ale unui limbaj de interogare.

- un utilizator interesat numai de o porțiune din baza de date poate evita contactul cu componentele ce nu prezintă interes;

Exemplu

Într-o bază de date cu două relații având schemele

ANGAJAT (NrAngajat, Departament)
MANAGER (Departament, NrSupervizor),

un utilizator interesat doar de angajați și de supervizorii lor ar putea fi ajutat de existența unei vederi definită astfel:

$$\pi_{NrAngajat, NrSupervizor} (ANGAJAT \bowtie MANAGER)$$