

Employee			Student		
EID	Name	Salary	SID	Name	Fees
1E	John	10,000	1S	Smith	1,000
2E	Ramesh	5,000	2S	Vijay	950
3E	Smith	8,000	3S	Gaurav	2,000
4E	Jack	6,000	4S	Nile	1,500
5E	Nile	15,000	5S	John	950

FIGURE 5.1. Employee and student relations.

Enostavne operacije:

Projekcija – oznaka Π :

$\Pi_S(R)$

Employee		
EID	Name	Salary
1E	John	10,000
2E	Ramesh	5,000
3E	Smith	8,000
4E	Jack	6,000
5E	Nile	15,000

Deluje na enojni relaciji R; vrne attribute, ki jih določimo s seznamom S

Eliminira duplicate.

Primer na tabeli Employee, Student

Če želimo vsa imena zaposlenih in njihove pláce iz relacije Employee:

$\Pi_{name, salary}$

Name	Salary
John	10,000
Ramesh	5,000
Smith	8,000
Jack	6,000
Nile	15,000

FIGURE 5.9. Result of projection operation.

SELEKCIJA:

Uporabljena zato, da se najde horizontalne podmnožice (tiste, ki zadostujejo pogoju) relacij na enojni relaciji. Uporablja se znak sigma.

Employee

EID	Name	Salary
1E	John	10,000
2E	Ramesh	5,000
3E	Smith	8,000
4E	Jack	6,000
5E	Nile	15,000

$\sigma_{\text{pogoj}}(R)$

Primer, če želimo, da imajo Employee plačo več kot 9000 v relaciji Employee

$\sigma_{\text{salary} > 9000}$

EID	Name	Salary
1E	John	10,000
5E	Nile	15,000

FIGURE 5.7. Result of selection operation.

rho
↓

PREIMENOVANJE:

$\rho_{S(S)}(R)$

R

A	B
1	5
2	3

$\rho_{S(C,D)}(R) \rightarrow$

S

C	D
1	5
2	3

S je novo ime relacije, S pa seznam novih imen atributov

OPERACIJE IZ TEORIJE MNOŽIC:

UNIJA:

Duplikati so izločeni, 2 nujna pogoja za union operacijo:

- a) Obe relaciji imata isto število atributov
- b) Tipi podatkov atributov morajo biti isti

RUS

Dve relaciji sta kompatibilni, če sledita zgornjima omejitvama

Primer: Če želiš najti imena vseh zaposlenih in imena vseh študentov skupaj

$\pi_{Name}(Employee) \cup \pi_{Name}(Student)$

Name
John
Ramesh
Smith
Jack
Nile
Vijay
Gaurav

FIGURE 5.2. Result of union operation.

PRESEK:

Uporabljena zato, da se najde skupne tuple med dvema relacijama. Isto morata držati zgornja pravila.

$R \cap S$

Primer: Če želiš najti vse zaposlene iz relacije Employee, ki so hkrati students.

$\Pi_{\text{Name}}(\text{Employee}) \cap \Pi_{\text{Name}}(\text{Student})$

Name
John
Smith
Nile

FIGURE 5.3. Result of set intersection operation.

RAZLIKA MNOŽIC:

Je binarna operacija, uporabljena, da se najde tuple, ki so prisotni v eni relaciji, ampak ne v drugi.

Uporablja se znak (-)

Primer: če želiš imena zaposlenih, kii niso študenti:

$\Pi_{\text{Name}}(\text{Employee}) - \Pi_{\text{Name}}(\text{Student})$

Name
Ramesh
Jack

FIGURE 5.4. Result of set-difference operation.

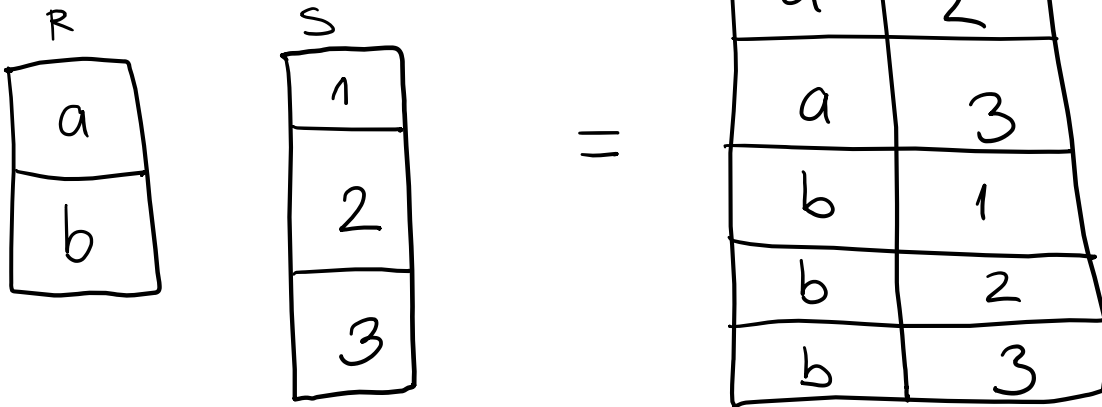
razlika relacij
 $R \cap S$
↓
 $R - S$

KARTEZIČNI PRODUKT:

Je binarna operacija, ki je uporabljena, da se združi informacije katerihkoli dveh operacij. Recimo, da ima relacij R1 m tuple in relacija R2 n tuple. Potem $R1 \times R2$ ima $m \times n$ tuplov.

$R \times S$

Primer:



(po domače: vsak z vsakim)

Employee			Job	
EID	Name	JID	JID	Job
1E	Manoj	1J	1J	Tester
2E	Deepak	2J	2J	Manager
3E	Vinay	1J		

FIGURE 5.5. Employee and Job relation.

EID	Name	Employee JID	Job JID	Job
1E	Manoj	1J	1J	Tester
1E	Manoj	1J	2J	Manager
2E	Deepak	2J	1J	Tester
2E	Deepak	2J	2J	Manager
3E	Vinay	1J	1J	Tester
3E	Vinay	1J	2J	Manager

FIGURE 5.6. Result of Cartesian product operation.

theta: $\theta \cup \theta$

POGOJNI STIK:

Enak kartezičnemu produktu, ampak ohranimo le n-terice, ki zadoščajo pogoju theta (θ)

$R \bowtie_{\theta} S$

Primer:

R

A	B
1	2
2	4

S

C	D
5	1
2	3

$R \bowtie_{B > D \wedge A = D}$

A	B	C	D
1	2	5	1
1	2	2	3
2	4	5	1
2	4	2	3

EKVISTIK:

Stik, kjer pogoj vsebuje le enakost

$R \bowtie_{\theta} S$

Primer:

R

A	B
1	2
3	4

S

C	D
3	2
1	4
5	3

$R \bowtie_{B=D} S$

A	B	C	D
1	2	3	2
1	2	1	4
1	2	5	3
3	4	3	2
3	4	1	4
3	4	5	3

NARAVNI JOIN:

Je ekvivalent po vseh skupnih atributih, ohranimo samo eno pojavitev skupnih atributov (ni podvojenih atributov), če pa relaciji nimata skupnih atributov, potem je to kartezični product

RMS

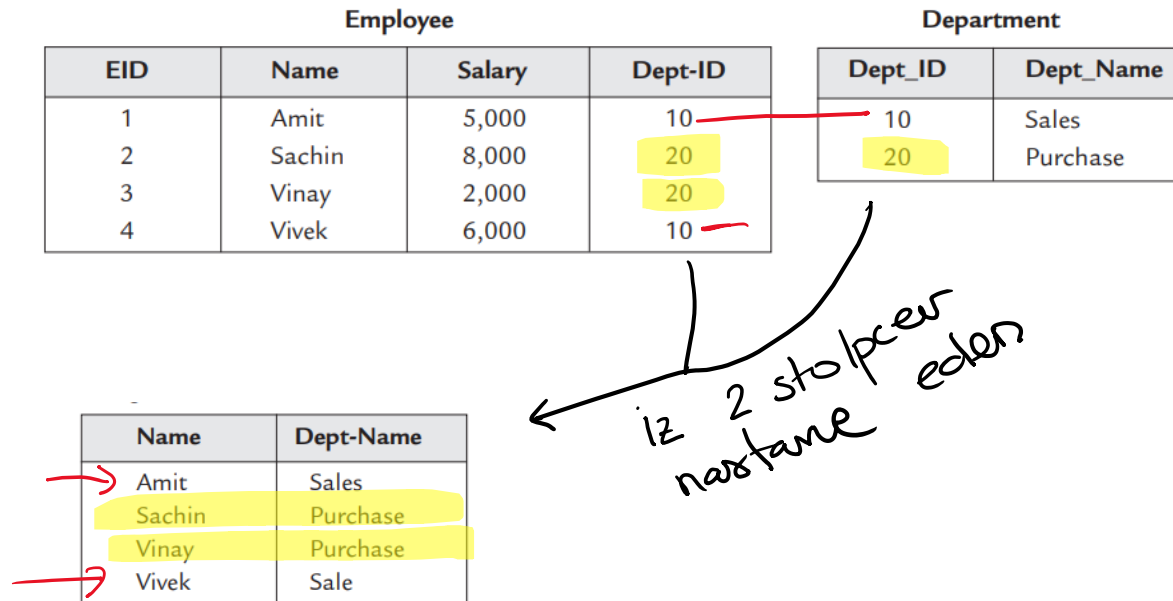


FIGURE 5.12. Result of natural join operation.

DELJENJE:

Obratno od kartezijskega produkta

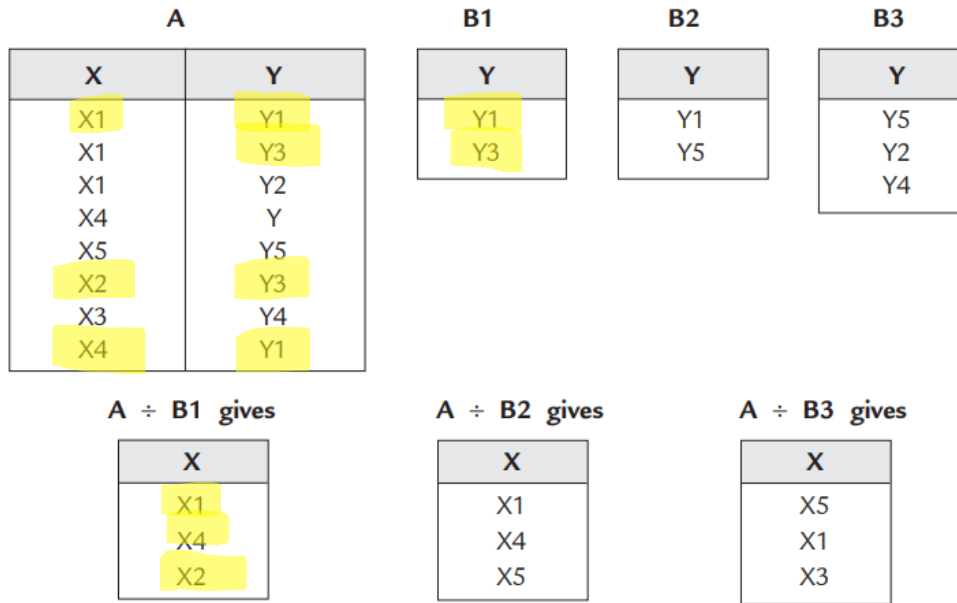


FIGURE 5.10. Result of division operation.

AGREGACIJA:

$$f_{AS}(R)$$

Seznam agregacijskih funkcij:

COUNT – prešteje vse ne-NULL vrednosti

SUM – sešteje vrednosti

AVG – izračuna povprečje

MIN – poišče najmanjšo vrednost

MAX – poišče največjo vrednost

Example. Consider the relation r as shown below:

A	B	C
α	α	17
α	β	20
β	β	13
β	β	30

1. $\mathcal{F}_{\text{sum}(c)}(r) = 80$ (i.e., $\text{sum}(C)=80$)
2. $\mathcal{F}_{\text{max}(c)}(r) = 30$ (i.e., $\text{max}(C)=30$)
3. $\mathcal{F}_{\text{min}(c)}(r) = 13$ (i.e., $\text{min}(C)=13$)

Primer:

R

Vpisna	Ocena	Ime
0001	7	Bob
0002	8	Ana
0001	7	Bob
0001	7	Bob

Vpisna, ime f_{AVG} OCENA (R)

Vpisna	Ime	...
0001	Bob	7
0002	Ana	8

VAJE LAŽJE: (vir: <https://tutorialwing.com/dbms-relational-algebra-examples-with-solutions/>)

Player relation

Player Id	Team Id	Country	Age	Runs	Wickets
1001	101	India	25	10000	300
1004	101	India	28	20000	200
1006	101	India	22	15000	150
1005	101	India	21	12000	400
1008	101	India	22	15000	150
1009	103	England	24	6000	90
1010	104	Australia	35	1300	0
1011	104	Australia	29	3530	10
1012	105	Pakistan	28	1421	166
1014	105	Pakistan	21	3599	205

a) Najdi vse tuple za katere je runs večji ali enak 15000

$$\sigma_{\text{runs} \geq "15000"}(\text{Player})$$

b) Izberi vse države v Player relaciji

$$\pi_{\text{country}}(\text{Player})$$

c) Izberi vse team ids in države v relaciji Player

$$\pi_{\text{TeamId, Country}}(\text{Player})$$

d) Najdi vse tuple iz player relacije za državo 'Indija'

$$\sigma_{\text{country}="India"}(\text{Player})$$

MALO ZAHTEVNEJŠE VAJE:

Vaja 1:

Recimo da imamo shemo:

Cities (CityCode, CityName, StateCode)

State (StateCode, StateName)

Opening (AccountNumber, OpenDate, OpeningBalance, TotalDeposit, TotalWithdrawal, ClosingBalance, ClosingBalanceDate, LastDepositDate, LastWithdrawalDate)

Deposit (AccountNumber, Date, Amount, Mode)

Withdrawal (AccountNumber, Date, Amount, Mode)

Account Holder (AccountNumber, Name, BuildingNumber, AreaName, StreetNumber, CityCode, PinCode, StateCode)

Zapiši:

- a) Naštej vse Account Number, ki niso imeli nič deposits in Jan, 2000

$\sigma_{Date > "2000-01-31" \wedge Date < "2000-01-01"} (\pi_{AccountNumber} (Deposit))$

↑ izberi datum (vrstice)

↑ izberi stolpec AccN.

↑ na relaciji Deposit

ali

$\pi_{AccountNumber} (\sigma_{Date > "2000-01-31" \wedge Date < "2000-01-01"} (Deposit))$

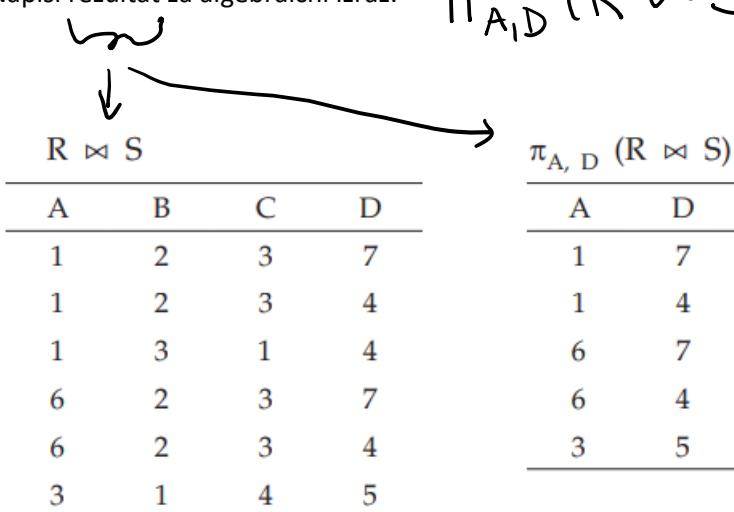
Vaja 2:

Imamo relaciji:

R		
A	B	C
1	2	3
1	2	0
1	3	1
6	2	3
1	4	2
3	1	4

S		
B	C	D
2	3	7
1	4	5
1	2	3
2	3	4
3	1	4

Napiši rezultat za algebraični izraz: $\pi_{A,D}(R \bowtie S)$



Vaja 3: (vir: <http://openclassroom.stanford.edu/MainFolder/courses/cs145/old-site/docs/backup/ra-exercises.html>)

Person (<u>name</u> , age, gender)	name is a key
Frequents (<u>name</u> , <u>pizzeria</u>)	(name, pizzeria) is a key
Eats (<u>name</u> , <u>pizza</u>)	(name, pizza) is a key
Serves (<u>pizzeria</u> , <u>pizza</u> , price)	(pizzeria, pizza) is a key

a) Poiščite vse picerije, ki jih obiskuje vsaj ena oseba, mlajša od 18 let.

$$\Pi_{\text{pizzeria}} (\sigma_{\text{age} < 18} (\text{Person}) \bowtie \text{Frequents})$$

b) Poiščite imena vseh žensk, ki jedo pico z gobami ali feferoni (ali oboje).

$$\Pi_{\text{name}} (\sigma_{\text{gender} = \text{"female"}} \wedge (\text{pizza} = \text{"mushroom"} \vee \text{pizza} = \text{"pepperoni"}) (\text{Person} \bowtie \text{Eats}))$$

c) Poiščite vse picerije, ki strežejo vsaj eno pico, ki jo Amy poje za manj kot 10,00 \$

$$\Pi_{\text{pizzeria}} (\sigma_{\text{name} = \text{"Amy"}} (\text{Eats}) \bowtie \sigma_{\text{price} < 10} (\text{Serves}))$$

Vaja 4: (<https://www.eecs.yorku.ca/~papagel/courses/eecs3421/docs/tutorials/tut1-ra.pdf>)

Suppliers(sID, sName, address)

Parts(pID, pName, colour)

Catalog(sID, pID, price)

Parts

id	Name	color
1	A	red
2	B	yellow
3	C	green

2. Find the names of all red parts.

Answer:

$\Pi_{pName}(\sigma_{colour="red"}(Parts))$

3. Find all prices for parts that are red or green. (A part may have different prices from different manufacturers.)

Answer:

$\Pi_{price}((\sigma_{colour="red" \vee colour="green"}(Parts)) \bowtie Catalog)$

zdruzil prek pID

4. Find the sIDs of all suppliers who supply a part that is red or green.

Answer:

$\Pi_{sID}((\sigma_{colour="red" \vee colour="green"}(Parts)) \bowtie Catalog)$

(isto kot zgornji primer)

5. Find the sIDs of all suppliers who supply a part that is red *and* green.

Answer: Trick question. Each tuple has only one colour, and each part has only one tuple (since pID is a key), so no part can be recorded as both red and green.

Catalog

sID	pID	Price	Name	Color
11	1	50	A	red
22	2	20	B	yellow
33	3	30	C	green