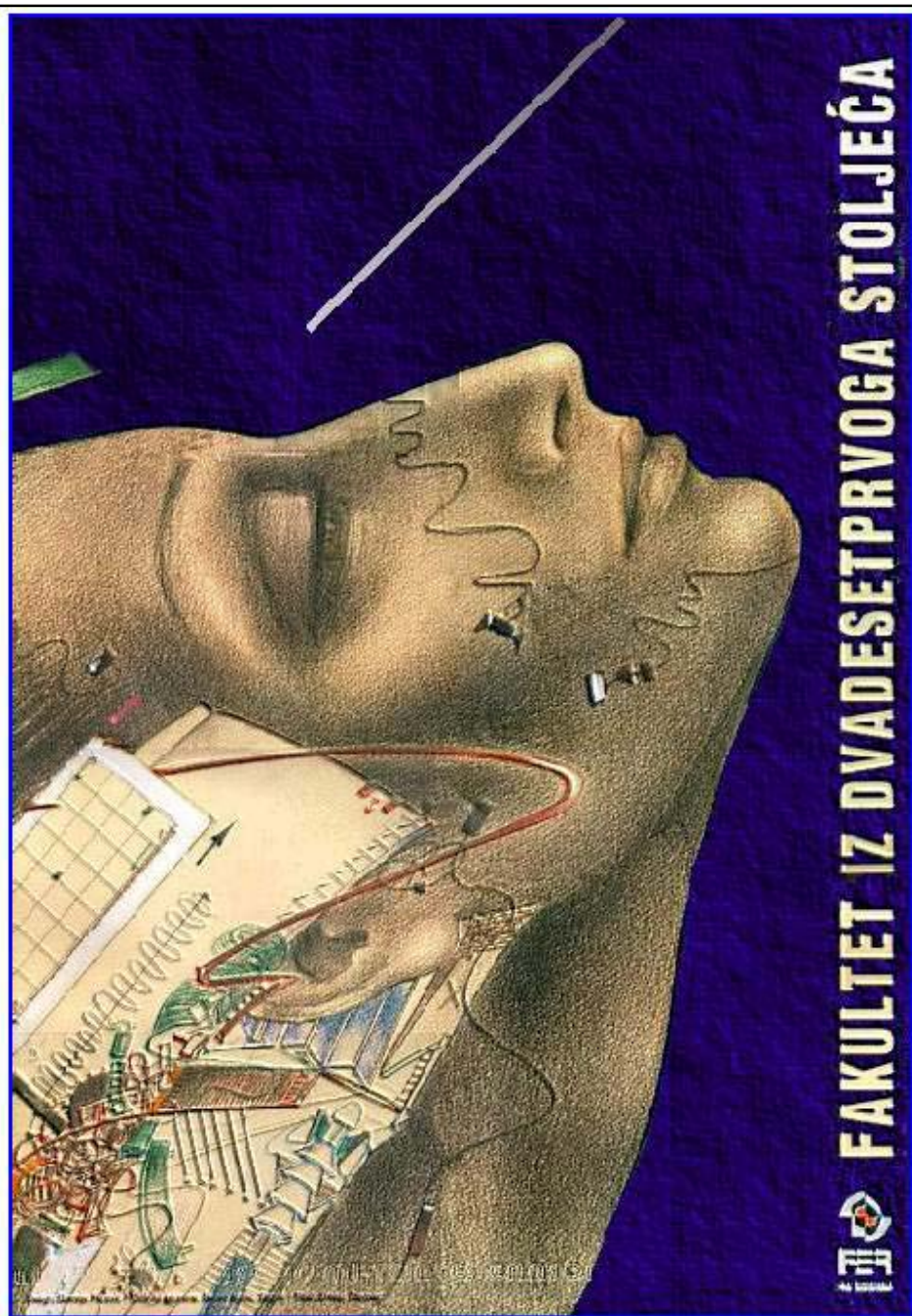


Baze podataka

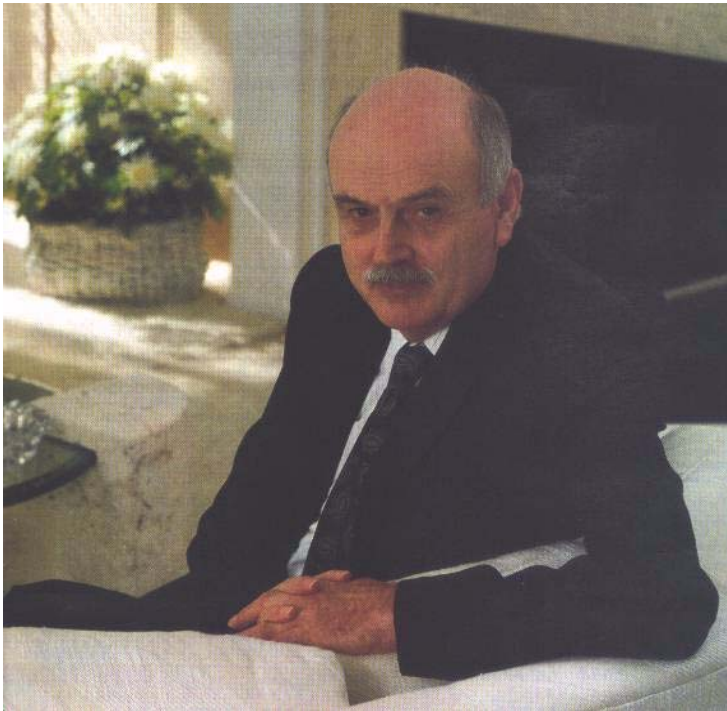
Predavanja
veljača 2008.

2. Relacijski model podataka



Relacijski model podataka

- E. F. Codd: "A Relational Model of Data for Large Shared Data Banks", Comm. ACM 13, No. 6, June 1970.



Dr. Edgar Frank Codd (1923-2003)

Ciljevi relacijskog modela podataka:

- osigurati visoki stupanj nezavisnosti podataka
- postaviti temelje za rješavanje problema semantike, konzistentnosti i redundancije podataka (**normalizacija**)
- omogućiti razvoj DML jezika temeljenih na operacijama nad skupovima

Relacijski model podataka

- Važni projekti u ranim 70-tim: jezik ISBL temeljen na relacijskoj algebri, jezici SQUARE i SEQUEL (DBMS System R) temeljeni na relacijskoj algebri i predikatnom računu te Query-By-Example temeljen na predikatnom računu nad domenama
 - razvojem prototipova dokazuje se praktična upotrebljivost relacijskog modela
 - postavljaju se temelji za rješavanje problema implementacije u područjima upravljanja transakcijama, paralelnog pristupa, obnove, optimizacije upita, sigurnosti i konzistentnosti podataka
- Projekti su potaknuli:
 - razvoj strukturiranog upitnog jezika (SQL)
 - razvoj komercijalnih **relacijskih** sustava za upravljanje bazama podataka (RDBMS)
 - Ingres, Oracle, IBM DB2, Informix, ...
 - danas: u upotrebi je nekoliko stotina različitih RDBMS sustava

Relacijski model podataka

- objekti u relacijskom modelu podataka su RELACIJE

mjesto

pbr	nazMjesto	sifZup
42000	Varaždin	7
51000	Rijeka	2
52100	Pula	4
51300	Delnice	2
42230	Ludbreg	7

zupanija

sifZup	nazZup
2	Primorsko-goranska
7	Varaždinska
4	Istarska

- **neformalna definicija:** relacija je imenovana dvodimenzionalna tablica
 - atribut je imenovani stupac relacije
 - domena je skup dopuštenih vrijednosti atributa
 - nad istom domenom može biti definiran jedan ili više atributa
 - n-torka (*tuple*) je redak relacije

Svojstva relacija

- relacija posjeduje ime koje je jedinstveno unutar sheme baze podataka
- atributi unutar relacije imaju jedinstvena imena
- jedan atribut može poprimiti vrijednost iz samo jedne domene
- u jednoj relaciji ne postoje dvije jednake n-torke
- redoslijed atributa unutar relacije je nebitan
- redoslijed n-torki unutar relacije je nebitan

zupanija

sifZup	nazZup
2	Primorsko-goranska
7	Varaždinska
4	Istarska

≡

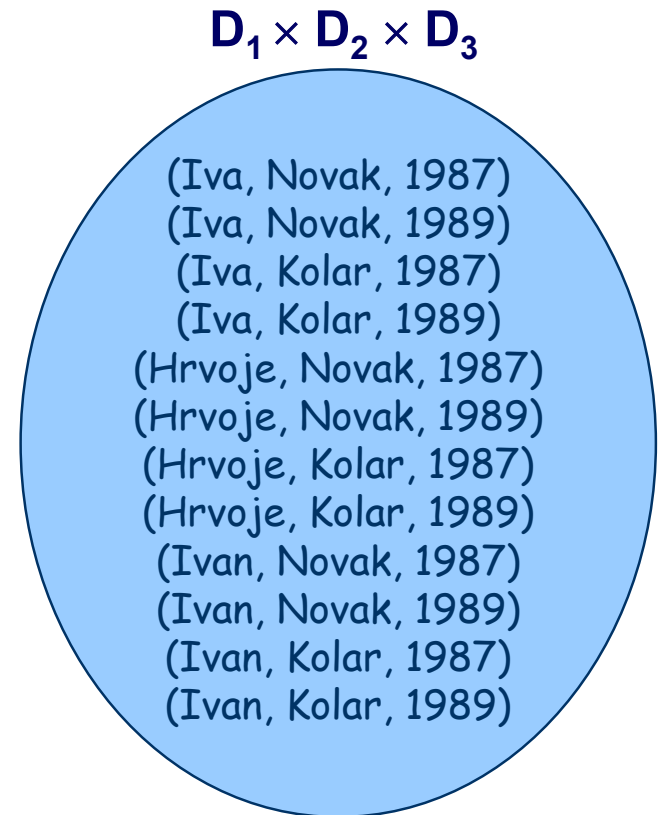
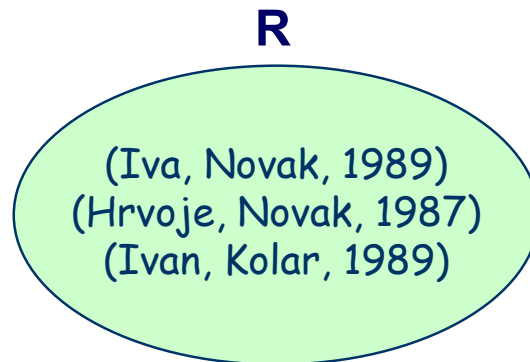
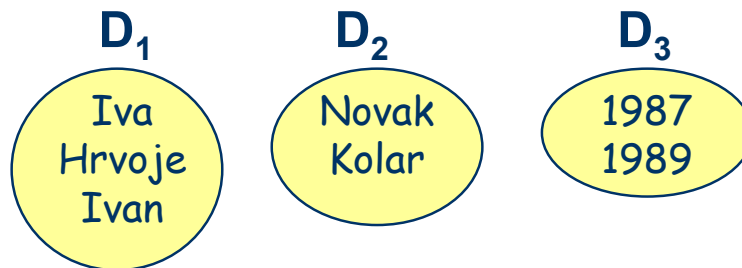
zupanija

nazZup	sifZup
Varaždinska	7
Istarska	4
Primorsko-goranska	2

Matematička relacija

- Relacija R definirana nad skupovima D_1, D_2, \dots, D_n je podskup Kartezijevog produkta skupova D_1, D_2, \dots, D_n

$$R \subseteq D_1 \times D_2 \times \dots \times D_n$$



Relacijska shema (formalna definicija)

- Neka su zadani atributi A_1, A_2, \dots, A_n . Relacijska shema R (intenzija) je imenovani skup atributa

$$R = \{ A_1, A_2, \dots, A_n \}$$

- radi pojednostavljenja, koristit će se i sljedeća notacija:

$$R = A_1 A_2 \dots A_n$$

- uočite: poredak atributa u shemi relacije je nebitan

$$R = \{ A_1, A_2, A_3 \} \equiv \{ A_3, A_1, A_2 \}$$

- Primjer: relacijska shema MJESTO

$$\text{MJESTO} = \{ \text{pbr}, \text{nazMjesto}, \text{sifZup} \}$$

Relacijska shema (primjer)

- Zadani su atributi pbr, nazMjesto, sifZup
- Relacijska shema
MJESTO = { pbr, nazMjesto, sifZup }
identična je relacijskoj shemi
MJESTO = { sifZup, pbr, nazMjesto }

n-torka (formalna definicija)

- Neka je $R = \{ A_1, A_2, \dots, A_n \}$ relacijska shema;
neka su D_1, D_2, \dots, D_n domene atributa A_1, A_2, \dots, A_n ;
n-torka t definirana na relacijskoj shemi R je skup parova oblika *atribut: vrijednostAtributa*

$$t = \{ A_1:v_1, A_2:v_2, \dots, A_n:v_n \},$$

pri čemu je $v_1 \in D_1, v_2 \in D_2, \dots, v_n \in D_n$

- Uočite: poredak elemenata n-torke nije bitan

$$\{ A_1:v_1, A_2:v_2, A_3:v_3 \} \equiv \{ A_3:v_3, A_1:v_1, A_2:v_2 \}$$

- Ponekad će se koristiti pojednostavljena notacija: pretpostavi li se da poredak vrijednosti atributa odgovara "poretku atributa" u relacijskoj shemi, n-torka se može prikazati na sljedeći način:

$$t = \langle v_1, v_2, \dots, v_n \rangle$$

n-torka (primjer)

- Zadana je relacijska shema OSOBA = { matBr, ime, prez }, pri čemu su domene atributa:

$\text{dom}(\text{matBr}) = \{1234, 1235, 1236, 1237\}$

$\text{dom}(\text{ime}) = \{\text{Iva}, \text{Hrvoje}, \text{Ivan}\}$

$\text{dom}(\text{prez}) = \{\text{Novak}, \text{Kolar}\}$

$t_1 = \{\text{matBr:1234}, \text{ime:Iva}, \text{prez:Novak}\}$

$t_2 = \{\text{matBr:1236}, \text{ime:Hrvoje}, \text{prez:Novak}\}$

$t_3 = \{\text{matBr:1237}, \text{ime:Ivan}, \text{prez:Kolar}\}$

- n-torka t_1 se jednako ispravno može napisati na sljedeći način (poredak elemenata n-torke je nebitan)

$t_1 = \{\text{ime:Iva}, \text{prez:Novak}, \text{matBr:1234}\}$

- pojednostavljena notacija:

$t_1 = \langle 1234, \text{Iva}, \text{Novak} \rangle$

Relacija (formalna definicija)

- Neka je $R = \{ A_1, A_2, \dots, A_n \}$ relacijska shema;
neka su D_1, D_2, \dots, D_n domene atributa A_1, A_2, \dots, A_n ;
relacija r (instanca relacije) definirana na shemi relacije R je skup n -torki koje su definirane na relacijskoj shemi R
- kad se želi naglasiti da je relacija r definirana na shemi relacije R , kao oznaka za relaciju koristi se
 $r(R)$ ili $r(\{ A_1, A_2, \dots, A_n \})$ ili $r(A_1 A_2 \dots A_n)$
- relacijska shema R : mijenja se relativno rijetko
- instanca relacije r : predstavlja trenutnu vrijednost relacije i često se mijenja (pri unosu/brisanju/izmjeni podataka)

Relacija (primjer)

- Zadana je relacijska shema $\text{STUDENT} = \{ \text{matBr}, \text{prez}, \text{slika} \}$, pri čemu su domene atributa:
 - $\text{dom}(\text{matBr}) = \{ 100, 102, 107, 111, 135 \}$
 - $\text{dom}(\text{prez}) = \{ \text{Novak}, \text{Kolar}, \text{Horvat}, \text{Ban} \}$
 - $\text{dom}(\text{slika}) = \{ \img alt="Illustration of a man in a blue shirt holding a clipboard" data-bbox="318 378 354 436" , \img alt="Illustration of a man in a plaid shirt holding a clipboard" data-bbox="388 378 424 436" , \img alt="Illustration of a woman in a white shirt holding a smartphone" data-bbox="444 378 488 436" } }$

$\text{student}(\text{STUDENT}) = \{ \{ \text{matBr:}102, \text{prez:Novak}, \text{slika:} \img alt="Illustration of a woman in a white shirt holding a smartphone" data-bbox="738 478 782 542" \}, \{ \text{matBr:}135, \text{prez:Ban}, \text{slika:} \img alt="Illustration of a man in a blue shirt holding a clipboard" data-bbox="712 542 752 602" \} \}$

- IDENTIČNA RELACIJA (poredak n-torki i članova n-torki je nebitan):

$\text{student}(\text{STUDENT}) = \{ \{ \text{prez:Ban}, \text{matBr:}135, \text{slika:} \img alt="Illustration of a man in a blue shirt holding a clipboard" data-bbox="712 712 752 772" \}, \{ \text{slika:} \img alt="Illustration of a woman in a white shirt holding a smartphone" data-bbox="448 772 492 832" , \text{matBr:}102, \text{prez:Novak} \} \}$

Relacija (primjer)

student(STUDENT) = { { prez:Ban, matBr:135, slika:  },
{ slika: , matBr:102, prez:Novak } }

- pojednostavljenje prikaza relacije (vizualizacija relacije tablicom)

student (STUDENT)

matBr	prez	slika
102	Novak	
135	Ban	

A-vrijednost n-torke, X-vrijednost n-torke

- Oznaka $t(A)$ predstavlja vrijednost koju atribut A poprima u n -torki t . $t(A)$ se naziva A -vrijednost n -torke t .

- Primjer:

$t = \{ \text{matBr:102, prez:Novak, slika:} \img alt="Illustration of a woman holding a smartphone" data-bbox="588 338 633 405"/> \}$
 $t(\text{prez}) = \text{Novak}$

- Neka je $X \subseteq R$. n -torka t reducirana na skup atributa X naziva se X -vrijednost n -torke t i označava s $t(X)$

- Primjer:

$t = \{ \text{matBr:102, prez:Novak, slika:} \img alt="Illustration of a woman holding a smartphone" data-bbox="595 735 640 802"/> \}$

$X = \{ \text{matBr, prez} \} \quad X \subseteq R$

$t(X) = t(\{ \text{matBr, prez} \}) = \{ \text{matBr:102, prez:Novak} \}$

Stupanj i kardinalnost relacije

- stupanj relacije: broj atributa (stupaca) - *degree*
- kardinalnost relacije: broj n-torki (redaka) - *cardinality*

mjesto

pbr	nazMjesto	sifZup
42000	Varaždin	7
51000	Rijeka	2
52100	Pula	4
51300	Delnice	2
42230	Ludbreg	7

kardinalnost = 5

stupanj = 3

Oznake:

$\text{deg}(\text{mjesto}) = 3$

$\text{card}(\text{mjesto}) = 5$

Shema i instanca baze podataka

- Shema baze podataka je skup relacijskih shema

$$\mathcal{R} = \{ R_1, R_2, \dots, R_n \}$$

- očito, relacijske sheme u jednoj shemi baze podataka moraju imati različita imena

- Instanca baze podataka definirana na shemi baze podataka $\mathcal{R} = \{ R_1, R_2, \dots, R_n \}$ je skup instanci relacija

$$r = \{ r_1(R_1), r_2(R_2), \dots, r_n(R_n) \}$$

- shema baze podataka se relativno rijetko mijenja
- instanca baze podataka se često mijenja

Operacije u relacijskom modelu podataka

Relacijska algebra

- Operacije relacijske algebre su:

\cup	unija (<i>union</i>)
\cap	presjek (<i>intersection</i>)
\setminus	razlika (<i>set difference</i>)
\div	dijeljenje (<i>division</i>)
π	projekcija (<i>projection</i>)
σ	selekcija (<i>selection</i>)
\times	Kartezijev produkt (<i>Cartesian product</i>)
ρ	preimenovanje (<i>renaming</i>)
\bowtie	spajanje (<i>join</i>)
	agregacija, grupiranje

Primjer: $r_4 = \sigma_{A=x \wedge B=y} (r_1 \cup (r_2 \cap r_3))$

- Karakteristika relacijske algebre - proceduralnost - navodi se redoslijed operacija koje se provode nad relacijama

Predikatni račun

- Operacije se specificiraju navođenjem predikata

$$r = \{ t \mid F(t) \}$$

- t je varijabla koja predstavlja:
 - n -torke - n -torski račun
 - rezultat r je skup n -torki t za koje je vrijednost predikata F istina
 - domene - domenski račun
 - rezultat je skup domena t za koje je vrijednost predikata F istina

- Primjer:

$$r_4 = \{ t \mid (r_1(t) \vee ((r_2(t) \wedge r_3(t)))) \wedge t(A)=x \wedge t(B)=y \}$$

- Predikatni račun je neproceduralan
 - ne navodi se redoslijed operacija
 - navode se predikati koje n -torke (domene) moraju zadovoljavati



SQL

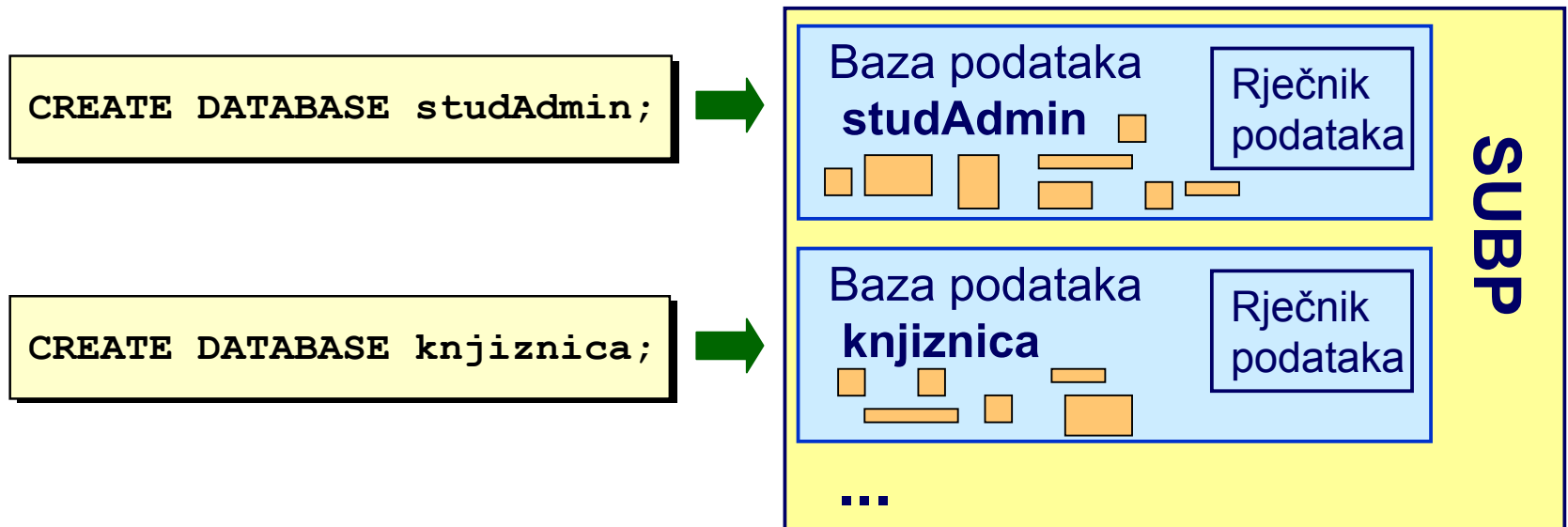
Kratki pogled

SQL - Kratki pogled

- SQL (*Structured Query Language*) je temeljen na relacijskom modelu podataka.
- nastao je na temelju jezika SEQUEL
- temelji se na predikatnom računu i relacijskoj algebri
- proglašen standardnim jezikom za relacijske sustave
- objekti u SQL-u su tablice, a ne (formalno definirane) relacije
 - poredak atributa (stupaca) u nekim je slučajevima značajan
 - u tablici ili rezultatu operacija nad tablicama moguća je pojava dvije ili više istih n-torki
 - ipak, postoje načini kako se to može spriječiti

SQL - Kratki pogled

- kreiranje nove instance baze podataka (kreiranje baze podataka)
 - jedan SUBP može istovremeno upravljati s više baza podataka



- Rječnik podataka sadrži opise relacijskih shema, integritetskih ograničenja, ...

`DROP DATABASE knjiznica;`



SQL - Kratki pogled

- opisivanje relacijske sheme (kreiranje relacije)
 - kreira praznu relaciju
 - ujedno je moguće definirati i integritetska ograničenja

```
CREATE TABLE mjesto (  
    pbr            INTEGER  
    , nazMjesto    CHAR(30)  
    , sifZup       SMALLINT  
);
```



mjesto		
pbr	nazMjesto	sifZup

```
DROP TABLE mjesto;
```



SQL - Kratki pogled

- upisivanje novih n-torki u relaciju

mjesto

pbr	nazMjesto	sifZup
-----	-----------	--------

```
INSERT INTO mjesto
VALUES (42000, 'Varaždin', 7);

INSERT INTO mjesto
VALUES (52100, 'Pula', 4);

INSERT INTO mjesto
VALUES (42230, 'Ludbreg', 7);
```



mjesto

pbr	nazMjesto	sifZup
42230	Ludbreg	7
42000	Varaždin	7
52100	Pula	4

Treba li poredak n-torki u relaciji biti u skladu s redoslijedom upisa?



SQL - Kratki pogled

- dohvat podataka iz relacije

mjesto

pbr	nazMjesto	sifZup
42230	Ludbreg	7
42000	Varaždin	7
52100	Pula	4

- dohvat podataka o mjestima čija šifra županije ima vrijednost 7

```
SELECT * FROM mesto  
WHERE sifZup = 7;
```



pbr	nazMjesto	sifZup
42000	Varaždin	7
42230	Ludbreg	7

SQL - Kratki pogled

- izmjena vrijednosti atributa u relaciji

mjesto

pbr	nazMjesto	sifZup
42230	Ludbreg	7
42000	Varaždin	7
52100	Pula	4

- naziv mjesta s poštanskim brojem 42000 promijeniti u VARAŽDIN

```
UPDATE mesto
  SET nazMjesto = 'VARAŽDIN'
  WHERE pbr = 42000;
```



mjesto

pbr	nazMjesto	sifZup
42230	Ludbreg	7
42000	VARAŽDIN	7
52100	Pula	4

SQL - Kratki pogled

- brisanje n-torki iz relacije

mjesto

pbr	nazMjesto	sifZup
42230	Ludbreg	7
42000	VARAŽDIN	7
52100	Pula	4

- obrisati mjesta za koje šifra županije ima vrijednost 7

```
DELETE FROM mesto  
WHERE sifZup = 7;
```



mjesto

pbr	nazMjesto	sifZup
52100	Pula	4



Relacijska algebra

Relacijska algebra

- **Unarne operacije**

- projekcija, selekcija, preimenovanje
- agregacija, grupiranje

- **Binarne operacije**

- skupovske operacije (*set operations*)
 - temelje se na relacijama kao skupovima n-torki
 - unija, presjek, razlika
- ostale binarne operacije
 - Kartezijev produkt, dijeljenje, spajanje

Relacijska algebra

- obavljanje operacije ne utječe na operande, npr.

$$r_3 = r_1 \cup r_2$$

- obavljanjem prethodne operacije nastaje nova relacija r_3 , a relacije r_1 i r_2 se pri tome ne mijenjaju
- operandi su relacije, a rezultat obavljanja operacije je uvijek relacija. To znači:
 - skup relacija je **zatvoren** s obzirom na operacije relacijske algebre
 - ta činjenica omogućava da se rezultat jedne operacije upotrijebi kao operand u sljedećoj operaciji, što omogućava formiranje složenih izraza

$$r_5 = (r_1 \cup r_2) \times (r_3 \bowtie r_4)$$

Unijska kompatibilnost

- Dvije relacije su unijski kompatibilne ukoliko vrijedi:
 - relacije su istog stupnja
 - i
 - korespondentni atributi su definirani nad istim domenama

polozioMatem

matBr	ime	prez
12345	Ivo	Kolar
13254	Ana	Horvat

polozioProgr

mbr	prezSt	imeSt
92632	Ban	Jura
67234	Novak	Iva

- relacije su istog stupnja
- $\text{dom}(\text{matBr}) = \text{dom}(\text{mbr})$
- $\text{dom}(\text{ime}) = \text{dom}(\text{imeSt})$
- $\text{dom}(\text{prez}) = \text{dom}(\text{prezSt})$

→ relacije su unijski kompatibilne

- kod ocjene jesu li relacije unijski kompatibilne
 - poredak atributa nije bitan
 - imena atributa nisu bitna

Unijska kompatibilnost

- dvije relacije koje imaju jednak broj atributa i jednaka imena atributa ne moraju ujedno biti unijski kompatibilne

zrakoplov		pecivo	
oznaka	naziv	oznaka	naziv
B-747	Boeing 747	ZE	Žemlja
A-360	Airbus 360	PR	Perec

- relacije su istog stupnja
- $\text{dom}(\text{zrakoplov.oznaka}) \neq \text{dom}(\text{pecivo.oznaka})$
- $\text{dom}(\text{zrakoplov.naziv}) \neq \text{dom}(\text{pecivo.naziv})$

→ relacije NISU unijski kompatibilne

- notacija *imeRelacije.imeAtributa* se često koristi kada je potrebno razlikovati istoimene attribute različitih relacija

Skupovske operacije: unija, presjek, razlika

- Skupovske operacije (unija, presjek, razlika) mogu se obavljati isključivo nad UNIJSKI KOMPATIBILNIM relacijama

Unija

- Rezultat operacije $r_1 \cup r_2$ je relacija čije su n-torke elementi relacije r_1 ili elementi relacije r_2 ili elementi obje relacije.
 - n-torke koje su elementi obje relacije u rezultatu se pojavljuju samo jednom (jer relacija je SKUP n-torki)

polozioMatem

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban
107	Jura	Horvat

polozioProgr

mbr	ime	prez
102	Ana	Novak
105	Rudi	Kolar
107	Jura	Horvat

polozioBaremJedan =
polozioMatem \cup polozioProgr

polozioBaremJedan

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban
105	Rudi	Kolar
107	Jura	Horvat

studenti koji su položili **ili**
Matematiku **ili**
Programiranje **ili**
oba predmeta

$$r_1 \cup r_2 \equiv r_2 \cup r_1$$

Presjek

- Rezultat operacije $r_1 \cap r_2$ je relacija čije su n-torke elementi relacije r_1 i elementi relacije r_2

polozioMatem

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban
107	Jura	Horvat

polozioProgr

mbr	ime	prez
102	Ana	Novak
105	Rudi	Kolar
107	Jura	Horvat

polozioOba =
polozioMatem \cap polozioProgr

polozioOba

mbr	ime	prez
102	Ana	Novak
107	Jura	Horvat

studenti koji su
položili i Matematiku
i Programiranje

$$r_1 \cap r_2 \equiv r_2 \cap r_1$$

Razlika

- Rezultat operacije $r_1 \setminus r_2$ je relacija čije su n-torke elementi relacije r_1 i nisu elementi relacije r_2

polozioMatem

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban
107	Jura	Horvat

polozioProgr

mbr	ime	prez
102	Ana	Novak
105	Rudi	Kolar
107	Jura	Horvat

**polozioSamoMatem =
polozioMatem \setminus polozioProgr**

polozioSamoMatem

mbr	ime	prez
100	Ivan	Kolar
103	Tea	Ban

studenti koji su
položili Matematiku,
ali **nisu** položili
Programiranje

$$r_1 \setminus r_2 \neq r_2 \setminus r_1$$

polozioSamoProgr = polozioProgr \setminus polozioMatem

polozioSamoProgr

mbr	ime	prez
105	Rudi	Kolar

ŠTO AKO SE IMENA KORESPONDENTNIH ATRIBUTA RAZLIKUJU

- Unija, presjek, razlika: u slučajevima kada su relacije unijski kompatibilne, ali se u relacijama koriste različita imena korespondentnih atributa, primjenjuje se sljedeći dogovor (konvencija): **kao imena atributa u rezultatnoj relaciji koriste se imena atributa prvog operanda**

polozioMatem

mbr	imeSt	prezSt
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban
107	Jura	Horvat

polozioProgr

mbr	ime	prez
102	Ana	Novak
105	Rudi	Kolar
107	Jura	Horvat

$$\text{polozioOba} = \text{polozioMatem} \cap \text{polozioProgr}$$

polozioOba

mbr	imeSt	prezSt
102	Ana	Novak
107	Jura	Horvat

Zadaci za vježbu

- zadane su unijski kompatibilne relacije
 - m (mbr ime prez) → studenti koji su položili Matematiku
 - d (mbr ime prez) → studenti koji su položili Dig. logiku
 - p (mbr ime prez) → studenti koji su položili Programiranje
- napisati izraze relacijske algebre koji određuju relacije koje sadrže studente (točnije rečeno n-torke):
 - a) koji su položili sva tri predmeta
 - b) koji su položili ili Matematiku ili Digitalnu logiku, ali ne oba predmeta (*ekskluzivni ili*)
 - c) koji su položili točno jedan (bilo koji) od ta tri predmeta
 - d) koji su položili bilo koja dva predmeta (ali nisu položili treći)

Dijeljenje (*division*)

- Zadane su relacije $r(R)$ i $s(S)$. Neka je $S \subseteq R$. Rezultat operacije $r \div s$ je relacija sa shemom $P = R \setminus S$. n -torka $t_r(P)$ se pojavljuje u rezultatu ako i samo ako za n -torku $t_r \in r$ vrijedi da se $t_r(P)$ u relaciji r pojavljuje u kombinaciji sa svakom n -torkom $t_s \in s$

polozen		predmet	
mbrSt	sifPred	sifPred	
100	1	1	2
100	2		
101	1		
101	2	3	3
101	3		
102	2		
102	3	1	2
103	1		
103	2		
103	3	3	3
104	3		

studenti koji su položili sve predmete
sa šiframa u relaciji predmet

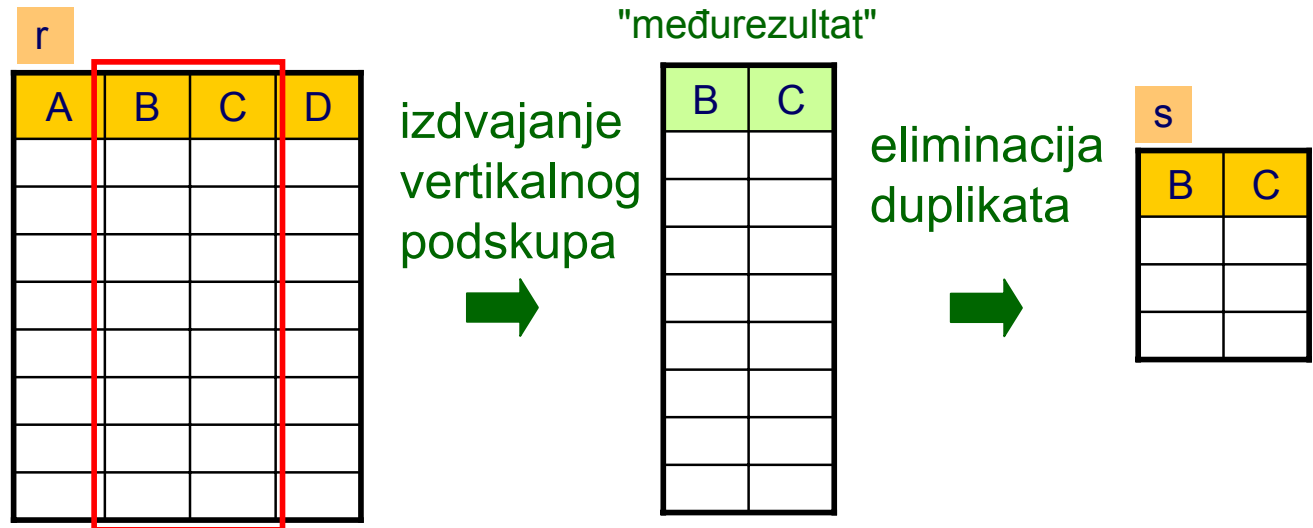
$\text{poloziliSve} = \text{polozen} \div \text{predmet}$

poloziliSve	
mbrSt	
101	
103	

Projekcija

- Zadana je relacija $r(R)$. Neka je skup atributa $\{ A_1, A_2, \dots, A_k \} \subseteq R$
- Obavljanjem operacije $\pi_{A_1, A_2, \dots, A_k}(r)$ dobiva se relacija s sa shemom $\{ A_1, A_2, \dots, A_k \}$ koja sadrži vertikalni podskup relacije r
 - $\text{deg}(s) = k$
 - $\text{card}(s) \leq \text{card}(r)$ (jer se eliminiraju duplikati)

$$s = \pi_{B, C}(r)$$



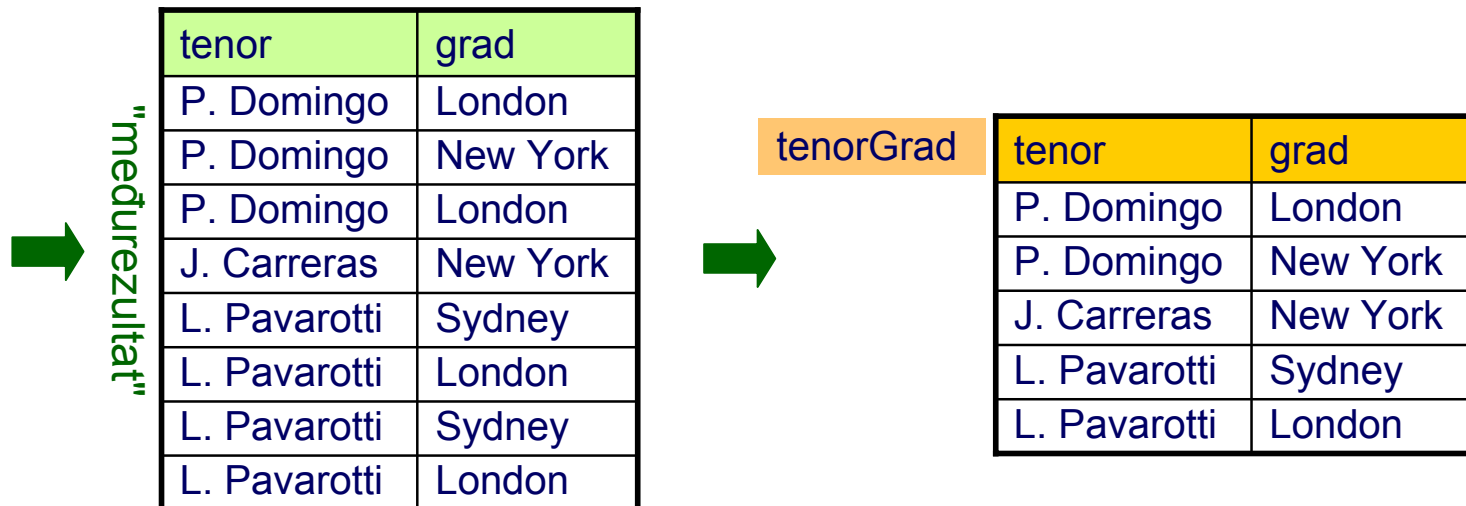
Projekcija (primjer)

Relacija nastup: u kojim gradovima **nastup** su nastupali koji tenori kojeg datuma

tenor	grad	datum
P. Domingo	London	15.2.1976
P. Domingo	New York	27.3.1981
P. Domingo	London	11.4.1987
J. Carreras	New York	11.4.1987
L. Pavarotti	Sydney	22.6.1992
L. Pavarotti	London	15.2.1976
L. Pavarotti	Sydney	19.1.1993
L. Pavarotti	London	14.7.1993

Traži se: u kojim gradovima su nastupali koji tenori

$$\text{tenorGrad} = \pi_{\text{tenor,grad}}(\text{nastup})$$



SQL - Lista za selekciju

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4

- **SELECT** *SELECT List* **FROM** *table*
- *SELECT List* je lista za selekciju: dio SELECT naredbe koji određuje koji će se "stupci" pojaviti u rezultatu

```
SELECT * FROM mjesto;
```

≡

```
SELECT mjesto.pbr
       , mjesto.nazMjesto
       , mjesto.sifZup
FROM mjesto;
```

- uz ime atributa može se navesti ime relacije (radi izbjegavanja dvosmislenosti u slučajevima kada se podaci dohvaćaju istovremeno iz više relacija čija se imena atributa podudaraju)
imeRelacije.imeAtributa
- u slučajevima kada takva dvosmislenost ne postoji, ime relacije se može (ali ne mora) ispustiti

SQL - Lista za selekciju

mjesto

pbr	nazMjesto	sifZup
42000	Varaždin	7
52100	Pula	4

zupanija

sifZupanija	nazZup
7	Varaždinska
4	Istarska

- u listi za selekciju se ne moraju navesti svi atributi relacije navedene u FROM dijelu naredbe:

```
SELECT nazMjesto  
      , pbr  
FROM mjesto;
```



nazMjesto	pbr
Varaždin	42000
Ludbreg	52100

- u listi za selekciju se mogu navesti samo oni atributi koji se nalaze u doseg SELECT naredbe, tj. atributi relacije koja je navedena u FROM dijelu naredbe:

```
SELECT nazMjesto  
      , pbr  
      , nazZup  
FROM mjesto;
```

Neispravna naredba

SQL - Projekcija

- za ispravno obavljanje projekcije nije dovoljno u listi za selekciju samo navesti imena atributa prema kojima se obavlja projekcija:

- primjer koji ujedno pokazuje kako rezultat SQL naredbe ne mora uvijek biti relacija



```
SELECT tenor  
      , grad  
FROM nastup;
```

Neispravna projekcija

tenor	grad
P. Domingo	London
P. Domingo	New York
P. Domingo	London
J. Carreras	New York
L. Pavarotti	Sydney
L. Pavarotti	London
L. Pavarotti	Sydney
L. Pavarotti	London

$\pi_{\text{tenor,grad}}(\text{nastup})$

```
SELECT DISTINCT tenor  
              , grad  
FROM nastup;
```

Ispravna projekcija

tenor	grad
P. Domingo	London
P. Domingo	New York
J. Carreras	New York
L. Pavarotti	Sydney
L. Pavarotti	London

Selekcija

- Zadana je relacija $r(R)$. Neka je F predikat (formula, uvjet, *condition*) koji se sastoji od operandada i operatora
 - operandi su:
 - imena atributa iz R
 - konstante
 - operatori su:
 - operatori usporedbe: $< \leq = \neq > \geq$
 - logički operatori: $\wedge \vee \neg$
- Obavljanjem operacije $\sigma_F(r)$ dobiva se relacija sa shemom R koja sadrži one n -torke relacije r za koje je vrijednost predikata F istina (*true*)

Selekcija (primjer)

student	matBr	ime	prez	postBr
	100	Ivan	Kolar	52000
	102	Ana	Horvat	10000
	105	Jura	Novak	21000
	107	Ana	Ban	51000

rezultat = $\sigma_{\text{ime} = \text{'Ana'} \vee \text{postBr} > 31000}$ (student)

- Za svaku pojedinu n-torku relacije:
 - vrijednosti atributa uvrštavaju se u predikat - uvrštavanjem vrijednosti u predikat dobiva se **sud**
 - onda i samo onda kada je vrijednost dobivenog suda istina (*true*), n-torka se pojavljuje u rezultatu selekcije

rezultat	matBr	ime	prez	postBr
	100	Ivan	Kolar	52000
	102	Ana	Horvat	10000
	107	Ana	Ban	51000

'Ivan' = 'Ana' \vee 52000 > 31000 \rightarrow *true*
'Ana' = 'Ana' \vee 10000 > 31000 \rightarrow *true*
'Jura' = 'Ana' \vee 21000 > 31000 \rightarrow *false*
'Ana' = 'Ana' \vee 51000 > 31000 \rightarrow *true*

SQL - Selekcija

- **SELECT** *SELECT List* **FROM** *table*
[WHERE Condition]
- Uvjet (*Condition*) se sastoji od operandada i operatora
 - operandi su:
 - imena atributa iz relacije *table*
 - konstante
 - operatori su:
 - operatori usporedbe: < <= = <> > >=
 - logički operatori: **AND** **OR** **NOT**
- Vrijednosti svake n-torke iz relacije *table* se uvrštavaju u *Condition* (a to je u stvari predikat). Ako je dobiveni sud istinit (*true*), n-torka se pojavljuje u rezultatu.

SQL - Selekcija

student

matBr	ime	prez	postBr
100	Ivan	Kolar	52000
102	Ana	Horvat	10000
105	Jura	Novak	21000
107	Ana	Ban	51000

$\sigma_{ime = 'Ana' \vee postBr > 31000} (student)$

```
SELECT * FROM student
WHERE ime = 'Ana'
      OR postBr > 31000;
```



matBr	ime	prez	postBr
100	Ivan	Kolar	52000
102	Ana	Horvat	10000
107	Ana	Ban	51000

SQL - Projekcija i selekcija

student	matBr	ime	prez	postBr
	100	Ivan	Kolar	52000
	102	Ana	Horvat	10000
	105	Jura	Novak	21000
	107	Ana	Ban	51000

$$\pi_{\text{ime}}(\sigma_{\text{ime} = \text{'Ana'} \vee \text{postBr} > 31000}(\text{student}))$$

```
SELECT DISTINCT ime
FROM student
WHERE ime = 'Ana'
      OR postBr > 31000;
```

"međurezultat"

matBr	ime	prez	postBr
100	Ivan	Kolar	52000
102	Ana	Horvat	10000
107	Ana	Ban	51000

ime
Ivan
Ana

Kartezijev produkt

- Zadana je relacija $r(R)$ i relacija $s(S)$, pri čemu je $R \cap S = \emptyset$.
- Obavljanjem operacije $r \times s$ dobiva se relacija $p(P)$, $P = R \cup S$.
n-torke relacije p se dobivaju spajanjem (ulančavanjem) svake n-torke iz relacije r sa svakom n-torkom iz relacije s
 - $\text{deg}(p) = \text{deg}(r) + \text{deg}(s)$
 - $\text{card}(p) = \text{card}(r) \cdot \text{card}(s)$

Kartezijev produkt (primjer)

student

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban

predmet

sifra	naziv
1	Programiranje
2	Matematika

upis = student \times predmet

upis

mbr	ime	prez	sifra	naziv
100	Ivan	Kolar	1	Programiranje
100	Ivan	Kolar	2	Matematika
102	Ana	Novak	1	Programiranje
102	Ana	Novak	2	Matematika
103	Tea	Ban	1	Programiranje
103	Tea	Ban	2	Matematika

SQL - Kartezijev produkt

- **SELECT** *SELECT List*
FROM *table* [, *table*] ...
[WHERE Condition]
- navede li se u FROM dijelu naredbe više od jedne relacije, obavlja se operacija Kartezijevog produkta navedenih relacija

student

mbr	ime	prez
100	Ivan	Kolar
102	Ana	Novak
103	Tea	Ban

predmet

sifra	naziv
1	Programiranje
2	Matematika

student × predmet

```
SELECT *  
FROM student, predmet;
```

```
SELECT student.*, predmet.*  
FROM student, predmet;
```

mbr	ime	prez	sifra	naziv
100	Ivan	Kolar	1	Programiranje
100	Ivan	Kolar	2	Matematika
102	Ana	Novak	1	Programiranje
102	Ana	Novak	2	Matematika
103	Tea	Ban	1	Programiranje
103	Tea	Ban	2	Matematika

SQL - Kartezijev produkt

- drugačija sintaksa:
- **SELECT *SELECT List***
FROM *table* [CROSS JOIN *table*]...
[WHERE *Condition*]

```
SELECT *  
FROM student CROSS JOIN predmet;
```

- Kartezijev produkt triju relacija:

```
SELECT *  
FROM r1 CROSS JOIN r2 CROSS JOIN r3;
```

Kartezijev produkt

- Što učiniti ukoliko je potrebno obaviti operaciju Kartezijevog produkta nad relacijama $r(R)$ i $s(S)$, u slučaju kada $R \cap S \neq \emptyset$

r	
A	B
1	a
2	b
3	c

s	
B	C
c	α
d	β

A	B	B	C
1	a	c	α
1	a	d	β
2	b	c	α
2	b	d	β
3	c	c	α
3	c	d	β


NIJE RELACIJA!

→ Potrebno je koristiti operaciju preimenovanja

Preimenovanje (relacije, atributa)

- Zadana je relacija $r(\{ A_1, A_2, \dots, A_n \})$
 - **preimenovanje relacije:** operacijom preimenovanja $\rho_s(r)$ dobiva se relacija **s** koja ima jednaku shemu i sadržaj kao relacija **r**
 - **preimenovanje relacije i atributa:** operacijom preimenovanja $\rho_{s(B_1, B_2, \dots, B_n)}(r)$ dobiva se relacija **s** čija shema umjesto atributa A_1, A_2, \dots, A_n sadrži attribute B_1, B_2, \dots, B_n , a sadržaj relacije **s** je jednak sadržaju relacije **r**

$$p = r \times \rho_{s(B_2, C)}(s)$$

r	A	B
	1	a
	2	b
	3	c

s	B	C
	c	α
	d	β

p	A	B	B2	C
	1	a	c	α
	1	a	d	β
	2	b	c	α
	2	b	d	β
	3	c	c	α
	3	c	d	β

SQL - Preimenovanje atributa

- ukoliko se drugačije ne navede, imena stupaca u rezultatu odgovaraju imenima atributa iz liste za selekciju
- implicitna imena stupaca rezultata se mogu promijeniti korištenjem operatora za preimenovanje **AS**

zupanija	sifZupanija	nazZup
	7	Varaždinska
	4	Istarska

```
SELECT sifZupanija AS sifraZ  
      , nazZup AS nazZ  
FROM zupanija;
```

sifraZ	nazZ
7	Varaždinska
4	Istarska

- rezervirana riječ AS smije se ispustiti

```
SELECT sifZupanija sifraZ  
      , nazZup nazZ  
FROM zupanija;
```

SQL - Preimenovanje atributa

- Primjer u kojem je potrebno koristiti preimenovanje atributa
 - SQL naredba bi bila ispravna i bez preimenovanja, ali tada kao rezultat ne bismo dobili relaciju (jer bi u shemi rezultata postojala dva atributa istog imena)

r	A	B
	1	a
	2	b
	3	c

s	B	C
	c	α
	d	β

$$r \times \rho_{s(B2, C)}(s)$$

```
SELECT A, r.B, s.B AS B2, C
FROM r, s;
```

A	B	B2	C
1	a	c	α
1	a	d	β
2	b	c	α
2	b	d	β
3	c	c	α
3	c	d	β

Spajanje uz uvjet ili θ - spajanje (θ - join)

- Zadane su relacije $r(R)$ i $s(S)$ pri čemu je $R \cap S = \emptyset$. Neka je F predikat oblika $r.A_i \theta s.B_j$, pri čemu je $A_i \in R$, $B_j \in S$, a θ je operator usporedbe iz skupa operatora $\{ <, \leq, =, \neq, >, \geq \}$
- Obavljanjem operacije $r \bowtie_F s$ dobiva se relacija koja sadrži n -torke iz $r \times s$ za koje je vrijednost predikata F istina (*true*), odnosno:

$$r \bowtie_F s = \sigma_F (r \times s)$$

što možemo reći o stupnju i kardinalnosti rezultata?

- Umjesto jednostavnog predikata $r.A_i \theta s.B_j$, može se koristiti složeni predikat dobiven primjenom logičkih operatora nad jednostavnim predikatima oblika $r.A_i \theta s.B_j$
- Problem spajanja uz uvjet relacija $r(R)$ i $s(S)$ kod kojih je $R \cap S \neq \emptyset$, rješava se na jednak način kao kod Kartezijevog produkta (korištenjem operatora preimenovanja)

Spajanje uz uvjet (primjer)

linija

let	udaljenost
CA-825	700
LH-412	4800
BA-722	15000
CA-311	13000

zrakoplov

tip	dolet
B747	13000
A320	5400
DC-9	3100

mogućnost = linija ▷◁ zrakoplov
dolet ≥ udaljenost

mogućnost

let	udaljenost	tip	dolet
CA-825	700	B747	13000
CA-825	700	A320	5400
CA-825	700	DC-9	3100
LH-412	4800	B747	13000
LH-412	4800	A320	5400
CA-311	13000	B747	13000



Linije i zrakoplovi koji na tim linijama mogu letjeti

SQL - Spajanje uz uvjet

- Koristi se ekvivalencija

$$r \bowtie_F s = \sigma_F (r \times s)$$

linija $\bowtie_{\text{dolet} \geq \text{udaljenost}}$ zrakoplov

```
SELECT *
```

```
FROM linija, zrakoplov
```

```
WHERE dolet >= udaljenost;
```

Kartezijev produkt

Selekcija

Linije i zrakoplovi koji na tim linijama mogu letjeti

linija

let	udaljenost
CA-825	700
LH-412	4800
BA-722	15000
CA-311	13000

zrakoplov

tip	dolet
B747	13000
A320	5400
DC-9	3100

let	udaljenost	tip	dolet
CA-825	700	B747	13000
CA-825	700	A320	5400
CA-825	700	DC-9	3100
LH-412	4800	B747	13000
LH-412	4800	A320	5400
CA-311	13000	B747	13000

SQL - Spajanje uz uvjet

- drugačija sintaksa:
- **SELECT** *SELECT List*
FROM table [JOIN table ON joinCondition] ...
[WHERE Condition]

```
SELECT *  
FROM linija JOIN zrakoplov  
ON dolet >= udaljenost;
```

- Spajanje uz uvjet triju relacija:

```
SELECT *  
FROM r1  
    JOIN r2  
        ON joinCondition  
    JOIN r3  
        ON joinCondition;
```

SQL - Spajanje uz uvjet i selekcija

- Kako pronaći linije i zrakoplove koji na tim linijama mogu letjeti, ali samo za one linije na kojima je udaljenost veća od 4000 km

$\sigma_{\text{udaljenost} > 4000}(\text{linija} \bowtie_{\text{dolet} \geq \text{udaljenost}} \text{zrakoplov})$

```
SELECT *  
FROM linija, zrakoplov  
WHERE dolet >= udaljenost  
AND udaljenost > 4000;
```

let	udaljenost	tip	dolet
LH-412	4800	B747	13000
LH-412	4800	A320	5400
CA-311	13000	B747	13000

ili

```
SELECT *  
FROM linija  
JOIN zrakoplov  
ON dolet >= udaljenost  
WHERE udaljenost > 4000;
```

SQL - Spajanje uz uvjet i projekcija

- Kako pronaći tipove zrakoplova koji se mogu iskoristiti za letove na postojećim linijama

$\pi_{\text{tip}}(\text{linija} \bowtie_{\text{dolet} \geq \text{udaljenost}} \text{zrakoplov})$

```
SELECT DISTINCT tip
FROM linija, zrakoplov
WHERE dolet >= udaljenost;
```

ili

```
SELECT DISTINCT tip
FROM linija
JOIN zrakoplov
ON dolet >= udaljenost;
```

tip
B747
A320
DC-9

Spajanje s izjednačavanjem (*Equi-join*)

- Spajanje relacija s izjednačavanjem je poseban oblik spajanja uz uvjet u kojem se kao θ operator koristi isključivo operator jednakosti (=)

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanija	sifZupanija	nazZup
	7	Varaždinska
	4	Istarska

mjestouZupaniji = mesto $\triangleright \triangleleft$ zupanija
sifZup = sifZupanija

mjestouZupaniji	pbr	nazMjesto	sifZup	sifZupanija	nazZup
	42000	Varaždin	7	7	Varaždinska
	52100	Pula	4	4	Istarska
	42230	Ludbreg	7	7	Varaždinska

- Problem spajanja s izjednačavanjem relacija $r(R)$ i $s(S)$ kod kojih je $R \cap S \neq \emptyset$, rješava se na jednak način kao kod Kartezijevog produkta (korištenjem operatora preimenovanja)

SQL - Spajanje s izjednačavanjem

- Koristi se ekvivalencija

$$r \bowtie_F s = \sigma_F (r \times s)$$

mjesto \bowtie zupanja
sifZup = sifZupanja

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanja	sifZupanja	nazZup
	7	Varaždinska
	4	Istarska

```
SELECT *  
  FROM mjesto, zupanja  
 WHERE sifZup = sifZupanja;
```

```
SELECT *  
  FROM mjesto  
        JOIN zupanja  
        ON sifZup = sifZupanja;
```

iii

SQL - Spajanje s izjednačavanjem

- U slučaju kada u relacijama postoje istoimeni atributi

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanija	sifZup	nazZup
	7	Varaždinska
	4	Istarska

mjesto ▷◁ zupanija

mjesto.sifZup = zupanija.sifZup

```
SELECT mjesto.*
      , zupanija.sifZup AS sifZup2
      , zupanija.nazZup
FROM mjesto, zupanija
WHERE mjesto.sifZup = zupanija.sifZup;
```

slično i u slučaju korištenja drugačije sintakse

Prirodno spajanje (*Natural Join*)

- Prirodno spajanje obavlja se na temelju jednakih vrijednosti istoimenih atributa.
- Zadane su relacije $r(R)$ i $s(S)$. Neka je $R \cap S = \{A_1, A_2, \dots, A_n\}$. Obavljanjem operacije $r \bowtie s$ dobiva se relacija sa shemom $R \cup S$ koja sadrži n -torke nastale spajanjem n -torki $t_r \in r, t_s \in s$, za koje vrijedi $t_r(A_1) = t_s(A_1) \wedge t_r(A_2) = t_s(A_2) \wedge \dots \wedge t_r(A_n) = t_s(A_n)$.

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanija	sifZup	nazZup
	7	Varaždinska
	4	Istarska

mjestouZupaniji = mesto \bowtie zupanija

mjestouZupaniji	pbr	nazMjesto	sifZup	nazZup
	42000	Varaždin	7	Varaždinska
	52100	Pula	4	Istarska
	42230	Ludbreg	7	Varaždinska

što možemo reći o stupnju rezultata?

Prirodno spajanje

- Rezultat prirodnog spajanje relacija $r(R)$ i $s(S)$ za koje vrijedi da je $R \cap S = \emptyset$ identičan je rezultatu obavljanja operacije Kartezijevog produkta $r \times s$

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanija	sifZupanija	nazZup
	7	Varaždinska
	4	Istarska

mjestouZupaniji = mesto $\triangleright \triangleleft$ zupanija

mjestouZupaniji	pbr	nazMjesto	sifZup	sifZupanija	nazZup
	42000	Varaždin	7	7	Varaždinska
	42000	Varaždin	7	4	Istarska
	52100	Pula	4	7	Varaždinska
	52100	Pula	4	4	Istarska
	42230	Ludbreg	7	7	Varaždinska
	42230	Ludbreg	7	4	Istarska

SQL - Prirodno spajanje

- prirodno spajanje se razlikuje od spajanja s izjednačavanjem po tome što se istoimeni atributi iz dviju relacija izbacuju (tako da od svakog ostane samo po jedan)

mjesto	pbr	nazMjesto	sifZup
	42000	Varaždin	7
	52100	Pula	4
	42230	Ludbreg	7

zupanija	sifZup	nazZup
	7	Varaždinska
	4	Istarska

```
SELECT mesto.*, zupanija.nazZup
FROM mesto, zupanija
WHERE mesto.sifZup = zupanija.sifZup;
```

pbr	nazMjesto	sifZup	nazZup
42000	Varaždin	7	Varaždinska
52100	Pula	4	Istarska
42230	Ludbreg	7	Varaždinska

SQL - Prirodno spajanje

- drugačija sintaksa:

```
SELECT mjesto.*, zupanija.nazZup  
FROM mjesto JOIN zupanija  
ON mjesto.sifZup = zupanija.sifZup;
```

Agregacija (*aggregation*)

ispit	mbrStud	akGod	nazPred	ocjena
	100	2005	Matematika	3
	101	2005	Matematika	5
	102	2005	Matematika	2
	103	2006	Matematika	3
	100	2004	Fizika	5
	101	2006	Fizika	5
	102	2006	Fizika	2
	100	2005	Vjerojatnost	4

- Kako izračunati prosjek ocjena na svim ispitima?

prosje	prosjeOcj
	3.625

Agregacija

- Zadana je relacija $r(R)$. Neka je atribut $A \in R$. Neka je \mathcal{AF} agregatna funkcija. Rezultat operacije agregacije $G_{\mathcal{AF}(A)}(r)$ je **relacija** stupnja 1 i kardinalnosti 1, pri čemu je vrijednost atributa određena primjenom funkcije \mathcal{AF} nad vrijednostima atributa A u svim n -torkama relacije r . Funkcija \mathcal{AF} može biti jedna od:
 - COUNT određuje broj pojava (broji sve, eventualni duplikati se također broje)
 - SUM izračunava sumu vrijednosti
 - AVG izračunava aritmetičku sredinu vrijednosti
 - MIN izračunava najmanju vrijednost
 - MAX izračunava najveću vrijednost
- naziv rezultatne relacije i atributa nije definiran operacijom, stoga se najčešće koristi u kombinaciji s operacijom preimenovanja
- također se koriste agregatne funkcije
 - COUNT-DISTINCT, SUM-DISTINCT, AVG-DISTINCT

Agregacija

ispit	mbrStud	akGod	nazPred	ocjena
	100	2005	Matematika	3
	101	2005	Matematika	5
	102	2005	Matematika	2
	103	2006	Matematika	3
	100	2004	Fizika	5
	101	2006	Fizika	5
	102	2006	Fizika	2
	100	2005	Vjerojatnost	4

- Prosjek ocjena na svim ispitima (rješenje):

$\rho_{\text{prosjek}(\text{prosjOcj})}(G_{\text{AVG}(\text{ocjena})(\text{ispit}))}$

prosjek	prosjOcj
	3.625

```
SELECT AVG(ocjena) AS prosjOcj  
FROM ispit;
```

prosjOcj
3.625

Agregacija (primjeri ostalih agregatnih funkcija)

osoba

sifra	tezina	visina
101	62	170
103	94	186
105	74	181
107	62	165

$\rho_{\text{rez1}(\text{broj1})}(G_{\text{COUNT}(\text{sifra})}(\text{osoba}))$

rez1

broj1
4

$\rho_{\text{rez2}(\text{broj2})}(G_{\text{SUM}(\text{tezina})}(\text{osoba}))$

rez2

broj2
292

$\rho_{\text{rez3}(\text{broj3})}(G_{\text{AVG}(\text{visina})}(\text{osoba}))$

rez3

broj3
175.5

$\rho_{\text{rez4}(\text{broj4})}(G_{\text{MAX}(\text{visina})}(\text{osoba}))$

rez4

broj4
186

$\rho_{\text{rez5}(\text{broj5})}(G_{\text{MIN}(\text{tezina})}(\text{osoba}))$

rez5

broj5
62

Moguće je odjednom izračunati više agregatnih vrijednosti:

$\rho_{\text{rez6}(\text{broj6}, \text{broj7}, \text{broj8})}(G_{\text{MIN}(\text{tezina}), \text{AVG}(\text{visina}), \text{MAX}(\text{visina})}(\text{osoba}))$

rez6

broj6	broj7	broj8
62	175.5	186

SQL - Agregatne funkcije

- naziv rezultatnog atributa nije definiran operacijom, stoga se koristi AS operator za preimenovanje

osoba	sifra	tezina	visina
	101	62	170
	103	94	186
	105	74	181
	107	62	165

```
SELECT COUNT(sifra) AS broj1 FROM osoba;
```

broj1

4

```
SELECT SUM(tezina) AS broj2 FROM osoba;
```

broj2

292

```
SELECT AVG(visina) AS broj3 FROM osoba;
```

broj3

175.5

```
SELECT MAX(visina) AS broj4,  
       MIN(tezina) AS broj5  
FROM osoba;
```

broj4

186

broj5

62

SQL - Agregatne funkcije

- agregatne funkcije s DISTINCT

osoba	sifra	tezina	visina
	101	62	170
	103	94	190
	105	74	170
	107	62	170

```
SELECT COUNT(DISTINCT visina) AS broj1  
FROM osoba;
```

broj1

2

```
SELECT SUM(DISTINCT tezina) AS broj2  
FROM osoba;
```

broj2

230

```
SELECT AVG(DISTINCT visina) AS broj3  
FROM osoba;
```

broj3

180

Agregacija i grupiranje

ispit	mbrStud	akGod	nazPred	ocjena
	100	2005	Matematika	3
	101	2005	Matematika	5
	102	2005	Matematika	2
	103	2006	Matematika	3
	100	2004	Fizika	5
	101	2006	Fizika	5
	102	2006	Fizika	2
	100	2005	Vjerojatnost	4

- Zadatak: izračunati prosječnu ocjenu za svaki pojedini predmet
 - prosjek za Matematiku
 - prosjek za Fiziku
 - ... i za sve ostale predmete čiji se naziv pojavljuje u relaciji

Agregacija i grupiranje

- Loše rješenje:

- Za svaki predmet napisati po jedan upit

$\rho_{\text{prosjeck(prosjOcjMat)}}(\mathcal{G}_{\text{AVG(ocjena)}}(\sigma_{\text{nazPred} = \text{'Matematika'}}(\text{ispit})))$

```
SELECT AVG(ocjena) AS prosjOcjMat
FROM ispit
WHERE nazPred = 'Matematika';
```

prosjaOcjMat
3.25

$\rho_{\text{prosjeck(prosjOcjFiz)}}(\mathcal{G}_{\text{AVG(ocjena)}}(\sigma_{\text{nazPred} = \text{'Fizika'}}(\text{ispit})))$

```
SELECT AVG(ocjena) AS prosjOcjFiz
FROM ispit
WHERE nazPred = 'Fizika';
```

prosjaOcjFiz
4

- itd. (za svaki naziv predmeta)
- postoji li bolje rješenje?

Grupiranje (*grouping*)

- Zadana je relacija $r(R)$. Neka su atributi $A_1, A_2, \dots, A_m, B_1, B_2, \dots, B_n$ atributi sheme R . Opći oblik operacije grupiranja je sljedeći:

$$A_1, A_2, \dots, A_m \mathcal{G}_{\mathcal{AF}_1(B_1), \mathcal{AF}_2(B_2), \dots, \mathcal{AF}_n(B_n)}(r)$$

- a) određuju se grupe n -torki: u svakoj grupi se nalaze n -torke koje imaju jednake vrijednosti atributa A_1, A_2, \dots, A_m
- b) za svaku grupu n -torki izračunavaju se vrijednosti agregatnih funkcija $\mathcal{AF}_1(B_1), \mathcal{AF}_2(B_2), \dots, \mathcal{AF}_n(B_n)$
- c) za svaku grupu formira se n -torka s vrijednostima atributa A_1, A_2, \dots, A_m i izračunatim vrijednostima agregatnih funkcija

Agregacija i grupiranje

ispit	mbrStud	akGod	nazPred	ocjena
	100	2005	Matematika	3
	101	2005	Matematika	5
	102	2005	Matematika	2
	103	2006	Matematika	3
	100	2004	Fizika	5
	101	2006	Fizika	5
	102	2006	Fizika	2
	100	2005	Vjerojatnost	4

- Za svaki predmet ispisati prosječnu ocjenu (ispravno rješenje):

$\rho_{\text{prosjeck}}(\text{nazPred}, \text{prosjeckOcj})_{(\text{nazPred } G_{\text{AVG}}(\text{ocjena}) (\text{ispit}))}$

- grupirati po nazPred
- za svaku grupu izračunati AVG(ocjena)
- za svaku grupu formirati po jednu n-torku s vrijednošću atributa nazPred i izračunatim prosjekom
- obaviti operaciju preimenovanja

prosjeck	
nazPred	prosjeckOcj
Matematika	4
Fizika	3.25
Vjerojatnost	4

Agregacija i grupiranje

- Ispisati prosječnu i najveću ocjenu za svaki predmet i akademsku godinu:

ispit	mbrStud	akGod	nazPred	ocjena
	100	2005	Matematika	3
	101	2005	Matematika	5
	102	2005	Matematika	2
	103	2006	Matematika	3
	100	2004	Fizika	5
	101	2006	Fizika	5
	102	2006	Fizika	2
	100	2005	Vjerojatnost	4

- u istu grupu ulaze n-torke koje imaju jednake vrijednosti atributa nazPred i akGod

$\rho_{\text{prosje1(nazPred, akGod, prosjOcj, maxOcj)}}(\text{nazPred, akGod } \mathcal{G}_{\text{AVG(ocjena), MAX(ocjena)}(\text{ispit}))$

prosje1	nazPred	akGod	prosjeOcj	maxOcj
	Matematika	2005	3.333	5
	Matematika	2006	3	3
	Fizika	2004	5	5
	Fizika	2006	3.5	5
	Vjerojatnost	2005	4	4

SQL - Grupiranje

- **SELECT** *SELECT List*
FROM ...
[WHERE *Condition*
[GROUP BY *column* [, *column*]**...]**

$\rho_{\text{prosjeck1(nazPred, akGod, prosjOcj, maxOcj)}}(\text{nazPred, akGod } G_{\text{AVG(ocjena), MAX(ocjena)}}(\text{ispit}))$

```
SELECT nazPred
      , akGod
      , AVG(ocjena) AS prosjOcj
      , MAX(ocjena) AS maxOcj
FROM ispit
GROUP BY nazPred, akGod;
```

nazPred	akGod	prosjaOcj	maxOcj
Matematika	2005	3.333	5
Matematika	2006	3	3
Fizika	2004	5	5
Fizika	2006	3.5	5
Vjerojatnost	2005	4	4

SQL - Grupiranje

- svi atributi koji se nalaze u listi za selekciju, a koji nisu argumenti agregatnih funkcija, moraju biti navedeni u GROUP BY dijelu naredbe

```
SELECT nazPred
      , akGod
      , mbrStud
      , AVG(ocjena) AS prosjOcj
      , MAX(ocjena) AS maxOcj
FROM ispit
GROUP BY nazPred, akGod;
```

NEISPRAVNO!

ispit

mbrStud	akGod	nazPred	ocjena
100	2005	Matematika	3
101	2005	Matematika	5
102	2005	Matematika	2
103	2006	Matematika	3
100	2004	Fizika	5
101	2006	Fizika	5
102	2006	Fizika	2
100	2005	Vjerojatnost	4

Zašto je to neispravno?

Za svaku grupu se formira samo po jedna n-torka: što s onim grupama u kojima postoji više vrijednosti atributa mbrStud?

nazPred	akGod	mbrStud	prosjaOcj	maxOcj
Matematika	2005	100, 101, 102 ?	3.333	5
Matematika	2006	?	3	3
Fizika	2004	?	5	5
Fizika	2006	?	3.5	5
Vjerojatnost	2005	?	4	4