

Baze podataka

Predavanja

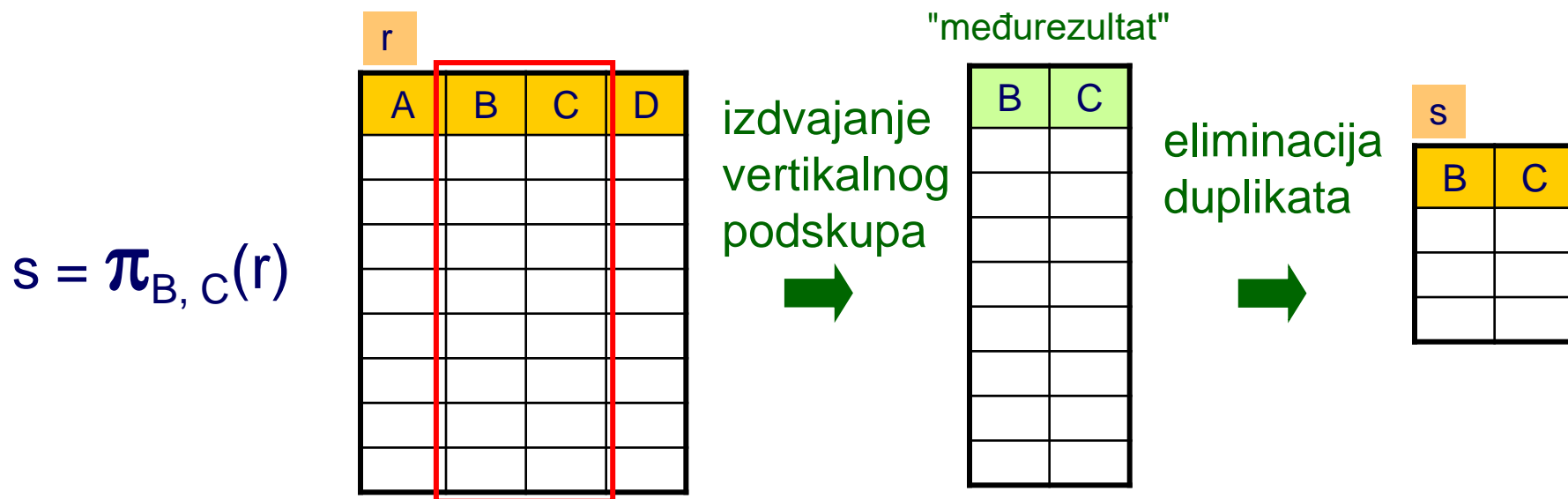
2. Relacijski model podataka – 2.dio

ožujak 2021.



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)




Projekcija (primjer)

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

Traži se: u kojim gradovima su nastupali koji tenori

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

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

 "meduerezultat"	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

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

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
Pula	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

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
 - $\text{deg}(p) = \text{deg}(r) + \text{deg}(s)$
 - $\text{card}(p) = \text{card}(r) \cdot \text{card}(s)$

Kartezihev produkt (primjer)

student

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

predmet

sifra	naziv
1	Programiranje
2	Matematika

$\text{upis} = \text{student} \times \text{predmet}$

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

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?

SQL - Prirodno spajanje

```
SELECT mjesto.*, zupanja.nazZup  
FROM mjesto, zupanja  
WHERE mjesto.sifZup = zupanja.sifZup;
```

- ili:

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


SQL - Prirodno spajanje – jednostavnija sintaksa

```
SELECT mjesto.*, zupanija.nazZup  
FROM mjesto JOIN zupanija  
    USING(sifzup);
```

■ ili:

```
SELECT mjesto.*, zupanija.nazZup  
FROM mjesto NATURAL JOIN zupanija;
```

NAPOMENA: ove naredbe specifične su za PostgreSQL!

Prirodno spajanje - relacije bez istoimenih atributa

- 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

Prirodno spajanje - relacije bez istoimenih atributa

- 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

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 $A_i \theta B_j$, 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 $A_i \theta B_j$, može se koristiti složeni predikat dobiven primjenom logičkih operatora nad jednostavnim predikatima oblika $A_i \theta B_j$
- 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 \geq 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 zrakoplov
dolet \geq udaljenost

```
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;
```

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

ili

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

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 \bowtie 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 zupanija
sifZup = sifZupanija

```
SELECT *  
  FROM mjesto, zupanija  
 WHERE sifZup = sifZupanija;
```

mjesto

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

zupanija

sifZupanija	nazZup
7	Varaždinska
4	Istarska

ili

```
SELECT *  
  FROM mjesto  
        JOIN zupanija  
        ON sifZup = sifZupanija;
```

SQL - Prirodno spajanje ≠ Spajanje s izjednačavanjem

- 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

mjestouZupaniji = mesto ▷◁ zupanija

```
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

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	prosOc
	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):

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

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

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

rez1	broj1
	4

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

rez2	broj2
	292

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

rez3	broj3
	175.5

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

rez4	broj4
	186

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

rez5	broj5
	62

Moguće je odjednom izračunati više agregatnih vrijednosti:

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

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

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

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

prosJocJMat
3.25

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

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

prosJocJFiz
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

$\rho_{\text{prosje}(nazPred, \text{pros}jOcj)}(nazPred \mathcal{G}_{AVG(ocjena)}(\text{ispit}))$

- 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

prosje	
nazPred	pros}Ocj
Matematika	3.25
Fizika	4
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

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

prosje1	nazPred	akGod	prosJocj	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*]...]

$\rho_{\text{prosjek1}(\text{nazPred}, \text{akGod}, \text{prosjOcj}, \text{maxOcj})}(\text{nazPred}, \text{akGod} \text{ } G_{\text{AVG}(\text{ocjena}), \text{MAX}(\text{ocjena})}(\text{ispit}))$

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

nazPred	akGod	prosjOcj	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

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