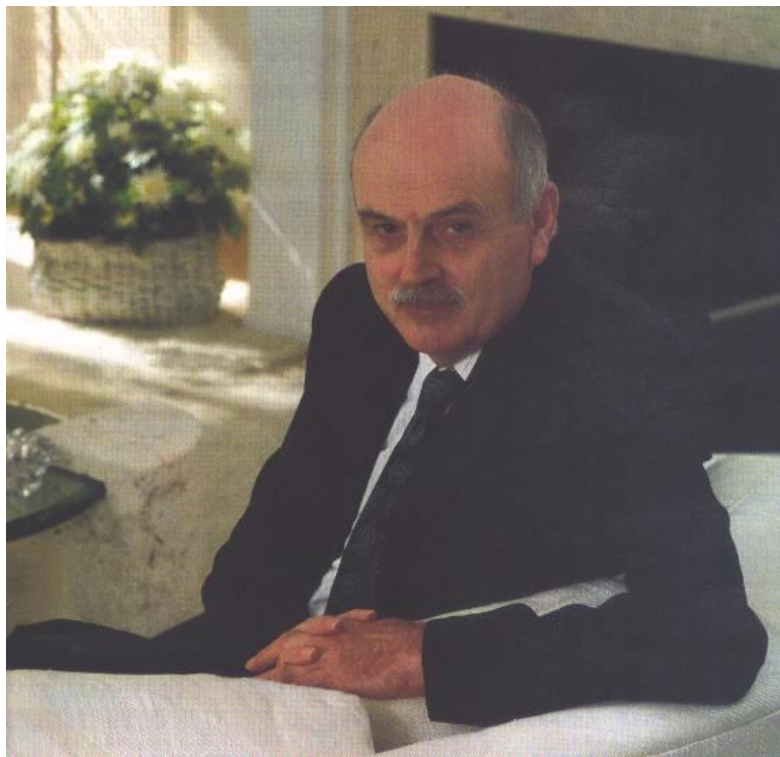




# Relacijski model podataka

- 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

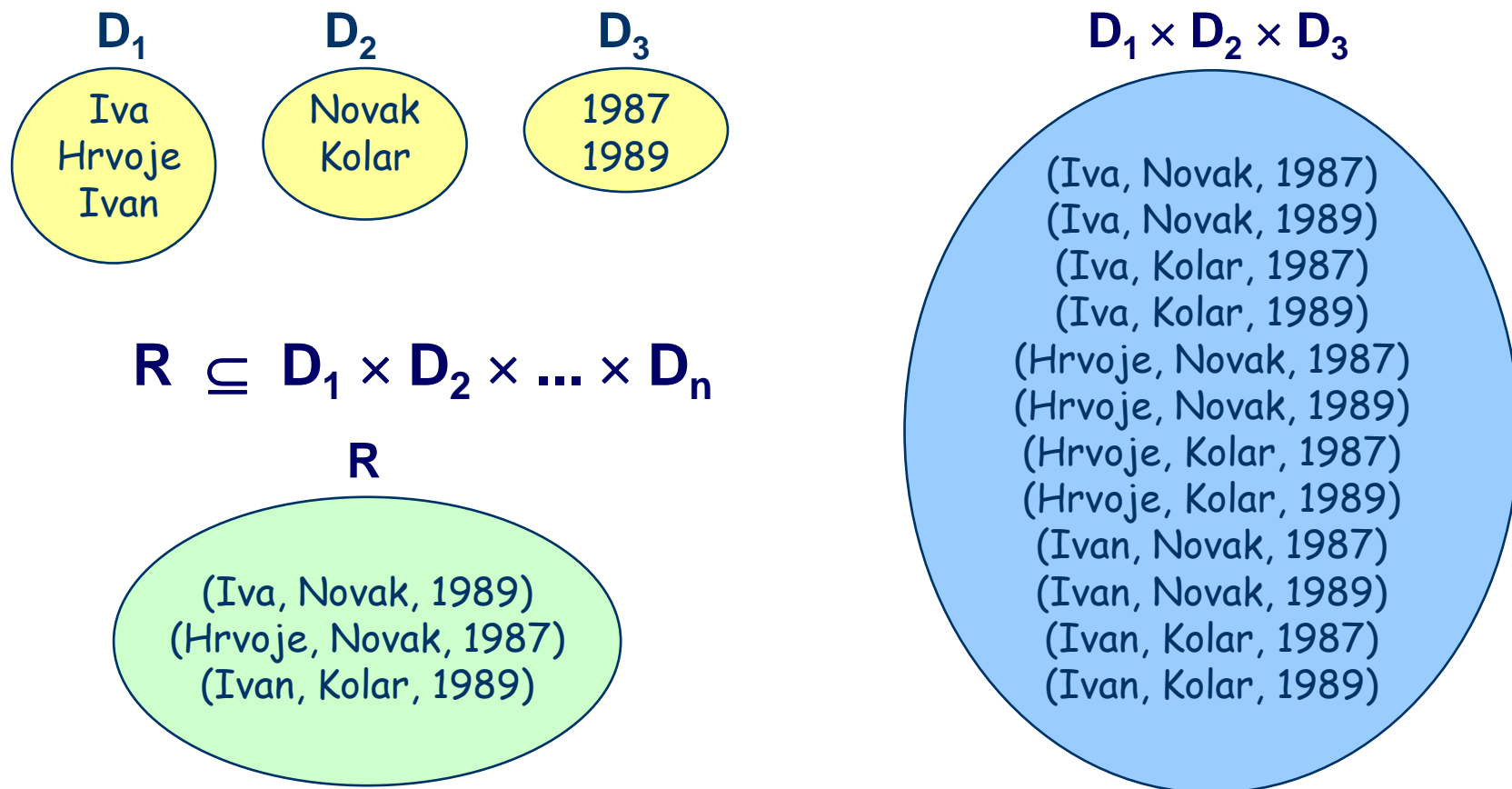
zupanija

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

# Matematička relacija

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



# 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: vrijednost* *Atributa*

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

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  
     $t_1 = \{\text{ime:Iva}, \text{prez:Novak}, \text{matBr:}1234\}$
- 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  $STUDENT = \{ matBr, prez, slika \}$ , pri čemu su domene atributa:
  - $dom(matBr) = \{ 100, 102, 107, 111, 135 \}$
  - $dom(prez) = \{ Novak, Kolar, Horvat, Ban \}$
  - $dom(slika) = \{  ,  ,  \}$

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

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

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

# Svojstva relacija

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

zupanija

sifZup	nazZup
2	Primorsko-goranska
7	Varaždinska
4	Istarska

≡



zupanija

nazZup	sifZup
Varaždinska	7
Istarska	4
Primorsko-goranska	2

# 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:

$t = \{ \text{matBr:102, prez:Novak, slika:} \img alt="Illustration of a person holding a smartphone" data-bbox="590 338 635 405"/> \}$

$t(\text{prez}) = \text{Novak}$

- 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:

$t = \{ \text{matBr:102, prez:Novak, slika:} \img alt="Illustration of a person holding a smartphone" data-bbox="598 738 643 805"/> \}$

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

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

# 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 (*union*)

$\cap$  presjek (*intersection*)

$\setminus$  razlika (*set difference*)

$\div$  dijeljenje (*division*)

$\pi$  projekcija (*projection*)

$\sigma$  selekcija (*selection*)

$\times$  Kartezijev produkt (*Cartesian product*)

$\rho$  preimenovanje (*renaming*)

$\bowtie$  spajanje (*join*)

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

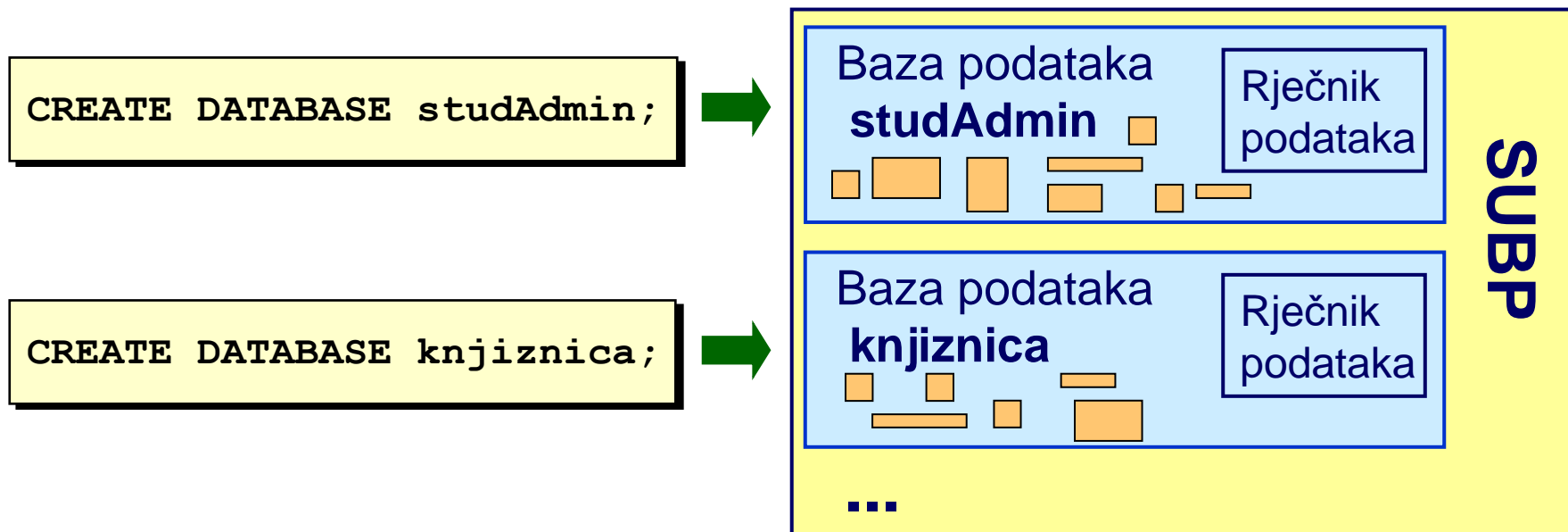
## Kratki pogled

# 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 mesto
VALUES (42000, 'Varaždin', 7);

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

INSERT INTO mesto
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

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

```
SELECT * FROM polozioMatem
UNION
SELECT * FROM polozioProgr;
```

# 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

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

```
SELECT * FROM polozioMatem
INTERSECT
SELECT * FROM polozioProgr;
```

# 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

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

**polozioSamoMatem = polozioMatem  $\setminus$  polozioProgr**

polozioSamoMatem

mbr	ime	prez
100	Ivan	Kolar
103	Tea	Ban

```
SELECT * FROM polozioMatem
EXCEPT
SELECT * FROM polozioProgr
```

**polozioSamoProgr = polozioProgr  $\setminus$  polozioMatem**

polozioSamoProgr

mbr	ime	prez
105	Rudi	Kolar

```
SELECT * FROM polozioProgr
EXCEPT
SELECT * FROM polozioMatem
```

# 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	
100	2	2	
101	1	3	
101	2		
101	3		
102	2		
102	3		
103	1		
103	2		
103	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	

# 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 operanada 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 - 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

<div>SELECT * FROM mesto;</div>	≡	<div>SELECT mesto.pbr       , mesto.nazMjesto       , mesto.sifZup FROM mesto;</div>
---------------------------------	---	--

- 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 - 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