



ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ
И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО



РЕЛАЦИОНА АЛГЕБРА И РЕЛАЦИОНИ ПРЕСМЕТКИ

БАЗИ НА ПОДАТОЦИ - предавања

Вон. проф. д-р Кире Триводалиев



Универзитет “Св. Кирил и Методиј” – Факултет за информатички науки и компјутерско инженерство

Преглед на лекцијата

➤ Релациона алгебра

➤ Единечни релациони операции

- SELECT (симбол: σ (sigma))
- PROJECT (симбол: π (pi))
- RENAME (симбол: ρ (rho))
- Генерализирана проекција

➤ Операции од релационата алгебра од теоријата на множества

- UNION (\cup), INTERSECTION (\cap), DIFFERENCE (или MINUS, $-$)
- ДЕКАРТОВ ПРОИЗВОД (\times)

➤ Бинарни релациони операции

- JOIN (потојат повеќе варијанти на JOIN)
- DIVISION

➤ Дополнителни релациони операции

- АГРЕГАТНИ ФУНКЦИИ (Пресметуваат сумарни операции: SUM, COUNT, AVG, MIN, MAX)
- OUTER JOINS, OUTER UNION

➤ Релациона пресметка

➤ Релациона пресметка на торка

➤ Релациона пресметка на домен

Преглед на релационата алгебра

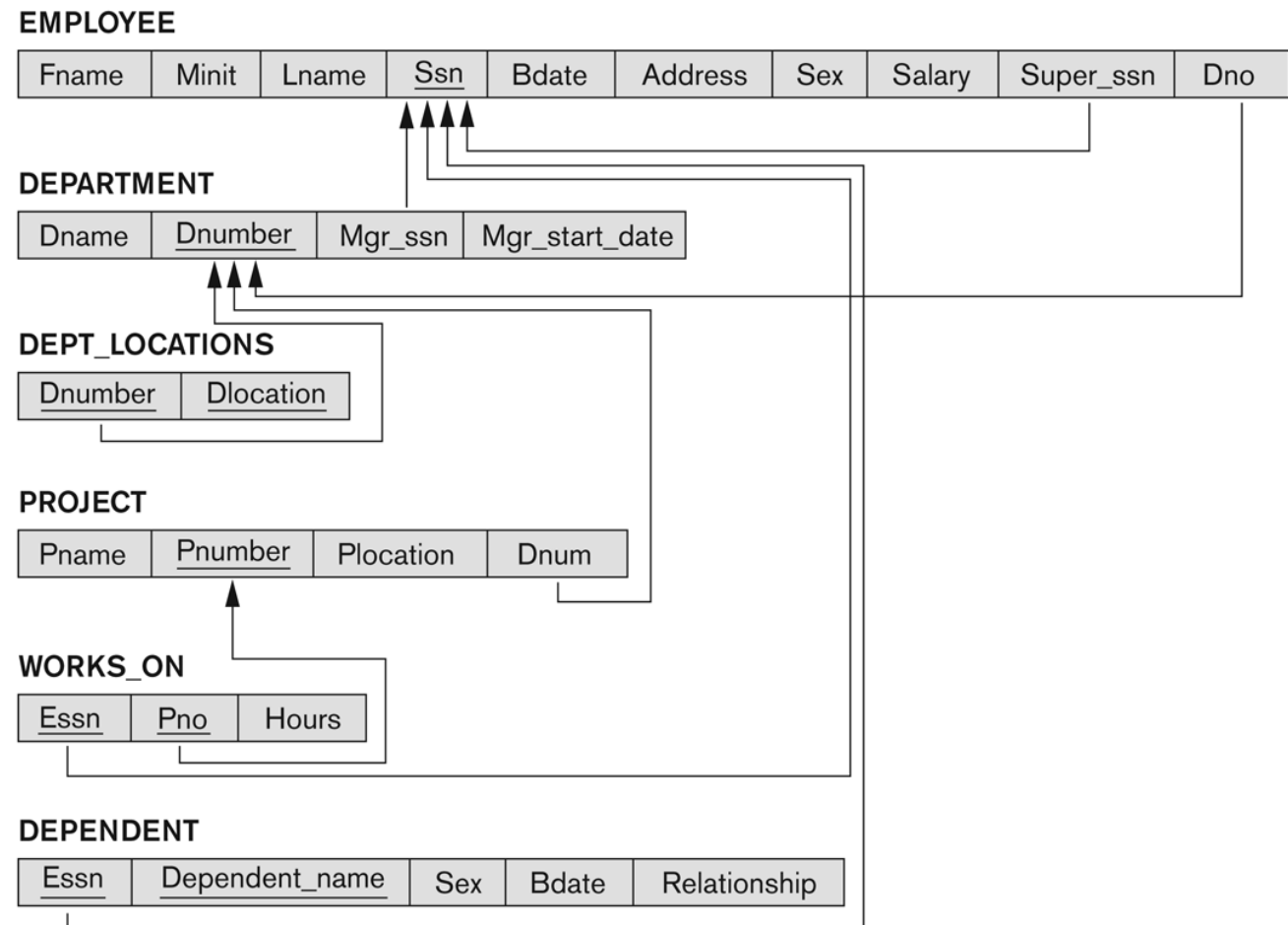
- Релационата алгебра го дава **основното множество операции** кои се применуваат над релациониот модел
- Овие операции дозволуваат корисникот да специфицира **основни барања за пребарување** (или **прашалници**)
- Резултатот од операцијата е **нова релација**, која може да биде формирана од една или повеќе **влезни** релации
 - Ова својство ја прави алгебрата “затворена” (сите објекти во релационата алгебра се релации)

Преглед на релационата алгебра (2)

- Со ова **операциите на алгебрата** произведуваат нови релации
 - Резултантните релации може дополнително да се обработуваат преку користење на операциите од истата алгебра
- Секвенцата од операции на релационата алгебра формира **израз во релационата алгебра**
 - Резултатот од изразот во релационата алгебра е исто така релација која претставува резултат на прашањето поставено кон базата на податоци (резултат од пребарувањето)

Шема на базата на податоци COMPANY

➤ Повеќето примери кои ќе се разгледуваат во текот на предавањето ќе се однесуваат на базата на податоци COMPANY претставена со следната релациска шема



Шема на базата на податоци COMPANY

- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)
- DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)
- DEPT_LOCATIONS(Dnumber*,Dlocations)
- PROJECT(Pname,Pnumber,Plocation,Dnum*)
- WORKS_ON(Essn*,Pno*,Hours)
- DEPENDENT(Essn*,Dependent_name,Sex,Bdate,Relationship)

Состојба на базата на податоци COMPANY

➤ Повеќето примери кои ќе се разгледуваат во текот на предавањето ќе се однесуваат на базата на податоци COMPANY претставена со следната релациска шема

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse



Единечна релациона операција SELECT

$$\sigma_{\langle \text{услов} \rangle}(R)$$

- SELECT операцијата (означена со σ (сигма)) се користи за селекција на *подмножество* од торките од кои е составена релацијата R преку користење на **условот за селекција**.
 - Условот за селекција се однесува како **филтер**
 - Релациони оператори: $<, \leq, \geq, >, =, \neq$
 - Изрази кои дефинираат услов:
 - $\langle \text{атрибут} \rangle$ оператор $\langle \text{константа} \rangle$, пр. $\text{Salary} > 30000$
 - $\langle \text{атрибут} \rangle$ оператор $\langle \text{атрибут} \rangle$, пр. $\text{Fname} = \text{Lname}$
 - $\langle \text{услов} \rangle$ AND $\langle \text{услов} \rangle$
 - $\langle \text{услов} \rangle$ OR $\langle \text{услов} \rangle$
 - NOT $\langle \text{услов} \rangle$
 - Торките кои го задоволуваат дефинираниот услов се задржуваат - *селектират*, додека пак останатите торки се *отфрлаат*
 - Резултатот кој се добива е нова релација со кратност иста како оригиналната релација

Единечна релациона операција SELECT



Пример

➤ Селектирај ги вработените кои работат во одделот со број 4:

Единечна релациона операција SELECT

Пример

- Селектирај ги вработените кои работат во одделот со број 4:
- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

Единечна релациона операција SELECT

Пример

- Селектирај ги вработените кои работат во одделот со број 4:
- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

$$\sigma_{Dno = 4} (EMPLOYEE)$$

Единечна релациона операција SELECT

Пример

- Селектирај ги вработените кои работат во одделот со број 4:
- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1



$\sigma_{Dno = 4} (EMPLOYEE)$

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4

Единечна релациона операција SELECT

Пример

- Селектирај ги вработените кои примаат плата поголема од \$30,000:
- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1



$\sigma_{\text{Salary} > 30000}(\text{EMPLOYEE})$

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

Единечна релациона операција SELECT

➤ Особини на SELECT операцијата

- SELECT операцијата $\sigma_{\langle \text{selection condition} \rangle}(R)$ продуцира нова релација S која ја има истата шема (истите атрибути) како и R
- SELECT σ е комутативна:
 - $\sigma_{\langle \text{condition1} \rangle}(\sigma_{\langle \text{condition2} \rangle}(R)) = \sigma_{\langle \text{condition2} \rangle}(\sigma_{\langle \text{condition1} \rangle}(R))$
- Поради комутативноста може да се примени секвенца на SELECT операции:
 - $\sigma_{\langle \text{cond1} \rangle}(\sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(R))) = \sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(\sigma_{\langle \text{cond1} \rangle}(R)))$
 - $\sigma_{\langle \text{cond1} \rangle}(\sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(R))) = \sigma_{\langle \text{cond1} \rangle \text{ AND } \langle \text{cond2} \rangle \text{ AND } \langle \text{cond3} \rangle}(R))$
- Бројот на торки во резултантната релација е помал (или еднаков со) бројот на торки во влезната релација R

Единечна релациона операција PROJECT

$$\pi_{\langle \text{листа на атрибути} \rangle}(R)$$

- Преку операцијата PROJECT (означена со π (pi)) се земаат одредени колони (атрибути) од релацијата R и се враќаат во резултантната релација
 - PROJECT креира вертикално партиционирање
 - Листата од специфицираните колони (атрибути) се чува за секоја торка
 - Останатите атрибути од торката се отфрлаат
 - Доколку во листата на атрибути **нема клучен атрибут**, тогаш во резултантната релација се отфрлаат дупликатите
 - теоријата на множества не дозволува појава на дупликат елементи

Единечна релациона операција PROJECT

Пример

- Да се излистаат името, презимето и платата на секој вработен (платен список):
- EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

$\pi_{\text{Lname, Fname, Salary}}(\text{EMPLOYEE})$

Единечна релациона операција PROJECT

Пример

- Да се излистаат името, презимето и платата на секој вработен (платен список):
- $\text{EMPLOYEE}(\text{Fname}, \text{Minit}, \text{Lname}, \underline{\text{Ssn}}, \text{Bdate}, \text{Address}, \text{Sex}, \text{Salary}, \text{Super_ssn}^*, \text{Dno}^*)$

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

$\pi_{\text{Lname}, \text{Fname}, \text{Salary}}(\text{EMPLOYEE})$



Fname	Lname	Salary
John	Smith	30000
Franklin	Wong	40000
Alicia	Zelaya	25000
Jennifer	Wallace	43000
Ramesh	Narayan	38000
Joyce	English	25000
Ahmad	Jabbar	25000
James	Borg	55000

Единечна релациона операција PROJECT

➤ Својства на операцијата PROJECT

- Бројот на торки во резултатот од проекцијата $\pi_{\langle \text{list} \rangle}(R)$ е секогаш помал или еднаков на бројот на торки во влезната релација R
- Операцијата PROJECT *не е* комутативна
 - $\pi_{\langle \text{list1} \rangle}(\pi_{\langle \text{list2} \rangle}(R)) = \pi_{\langle \text{list1} \rangle}(R)$
 - Горното е можно ако и само ако $\langle \text{list2} \rangle$ ги содржи атрибутите дефинирани во $\langle \text{list1} \rangle$

Примери на користење на σ и π

Пример

- Најди ги вработените кои имаат плата поголема од 25000 од четвртиот оддел или кои имаат плата поголема од 30000 од петтиот оддел?
- $\text{EMPLOYEE}(\text{Fname}, \text{Minit}, \text{Lname}, \underline{\text{Ssn}}, \text{Bdate}, \text{Address}, \text{Sex}, \text{Salary}, \text{Super_ssn}^*, \text{Dno}^*)$

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1



$\sigma_{(\text{Dno}=4 \text{ AND } \text{Salary} > 25000) \text{ OR } (\text{Dno}=5 \text{ AND } \text{Salary} > 30000)} (\text{EMPLOYEE})$

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5

Примери на користење на σ и π

- Врати ги полот и платата на сите вработени.
- $\text{EMPLOYEE}(\text{Fname}, \text{Minit}, \text{Lname}, \underline{\text{Ssn}}, \text{Bdate}, \text{Address}, \text{Sex}, \text{Salary}, \text{Super_ssn}^*, \text{Dno}^*)$

Пример

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1



$\pi_{\text{Sex}, \text{Salary}}(\text{EMPLOYEE})$

Sex	Salary
M	30000
M	40000
F	25000
F	43000
M	38000
M	25000
M	55000

Изрази во релационата алгебра

- Возможно е да се применат неколку операции од релационата алгебра едно после друго, со што се добиваат **изрази** од релационата алгебра
 - Сите операции може да се вгнездат една во друга, со цел да се добие **единечен израз од релационата алгебра**, или
 - Може да се користи принципот на **меѓу резултантни релации**, каде во еден момент се извршува само една операција од релационата алгебра
 - Во овој случај, мора да се специфицираат имиња на меѓу-резултантните релации, со цел истите да се користат како влезни релации во следните изрази
 - Се користи операторот за доделување ←

Изрази во релационата алгебра

Пример

➤ Да се добијат имињата и презимињата и платите на вработените кои работат во одделот со број 5?

➤ Може да се напише *единечен израз од релационата алгебра*:

$$\pi_{\text{Fname, Lname, Salary}}(\sigma_{\text{DNO}=5}(\text{EMPLOYEE}))$$

➤ Резултатот е

Fname	Lname	Salary
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

Изрази во релационата алгебра

Пример

- Да се добијат имињата и презимињата и платите на вработените кои работат во одделот со број 5?
- Или може да се даде секвенца од операции, при што мора да се специфицира име на меѓу-релацијата која се добива во секој чекор:

$$\text{DEP5_EMPS} \leftarrow \sigma_{\text{Dno}=5}(\text{EMPLOYEE})$$

$$\text{RESULT} \leftarrow \pi_{\text{Fname, Lname, Salary}}(\text{DEP5_EMPS})$$

- Резултатот е

DEP5_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

RESULT

Fname	Lname	Salary
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

Единечна релациона операција RENAME

$$\rho_S (B_1, B_2, \dots, B_n)(R)$$

- Операторот RENAME (се означува со симболот ρ (rho)) се користи во ситуации кога сакаме да ги **промениме** имињата на атрибутите од релацијата или да го промениме името на самата релација или и двете
 - Корисно кога прашалникот бара повеќе операции
 - Неопходно во некои случаи (на пример во JOIN операторот со кој ќе се запознаеме подоцна)

Единечна релациона операција RENAME

- Нека е дадена релацијата $R (A_1, A_2, \dots, A_n)$
- Во општ случај операцијата RENAME може да се пронајде во следните форми:
 - $\rho_{S (B_1, B_2, \dots, B_n)}(R)$ ги менува:
 - името на релацијата во S , и
 - имињата на колоните (атрибутите) во B_1, B_2, \dots, B_n
 - $\rho_S(R)$ го менува:
 - само името на релацијата во S
 - $\rho_{(B_1, B_2, \dots, B_n)}(R)$ ги менува:
 - имињата на колоните (атрибутите) во B_1, B_2, \dots, B_n

Единечна релациона операција RENAME

- Во некои случаи многу позгодно е да користиме кратенки за имињата на атрибутите отколку самите имиња на атрибути:

$$\text{RESULT} \leftarrow \pi_{\text{Fname, Lname, Salary}}(\text{DEP5_EMPS})$$

RESULT ќе ги има *истите имиња на атрибутите* дефинирани во DEP5_EMPS (истите атрибути како EMPLOYEE)

$$\text{RESULT} \leftarrow \rho_{(\text{F,M,L,S,B,A,SX,SAL,SU,DNO})}(\text{DEP5_EMPS})$$

10-те атрибути на DEP5_EMPS ќе се *променат* во F, M, L, S, B, A, SX, SAL, SU, DNO, соодветно

DEP5_EMPS

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

Генерализирана проекција

$$\pi_{\langle \text{листа на функции над атрибути} \rangle}(R)$$

- $\langle \text{листа на функции над атрибути} \rangle = F1, F2, \dots, F_n$
 - $F1, F2, \dots, F_n$ се функции над атрибутите на релацијата R и може да вклучуваат аритметички операции и константи
 - Многу корисна операција кога е потребно во резултатот да се вратат вредности кои се пресметуваат врз основа на вредности на атрибути во некоја релација

Пример на користење на генерализирана проекција

Пример

- Нека е дадена модифицирана релација за вработени со следната дефиниција
 $EMPLOYEE_S(Ssn, Salary, Deduction, Years_service)$
 каде Salary е бруто плата, Deduction се задолжителни придонеси, и Years_service е години на стаж
- Да се врати список на сите вработени заедно со нето платата која ја земаат и бонус кој го добиваат ако важи дека:
 - Нето плата = Бруто плата – Придонеси
 - Бонус = Години стаж * 1000

$$\pi_{Ssn, Salary-Deduction, Years_service*1000}(EMPLOYEE_S)$$

Потенцијален проблем е како се именуваат атрибутите кои се добиваат во резултатот?

Пример на користење на генерализирана проекција

$$\pi_{\text{Ssn, Salary-Deduction, Years_service*1000}} (\text{EMPLOYEE_S})$$

Пример

Потенцијален проблем е како се именуваат атрибутите кои се добиваат во резултатот?

Решение 1 со користење на оператор за доделување:

$$\text{RESULT}(\text{Ssn, Net_Salary, Bonus}) \leftarrow \pi_{\text{Ssn, Salary-Deduction, Years_service*1000}} (\text{EMPLOYEE_S})$$

Решение 2 со користење на операцијата RENAME:

$$\text{RESULT} \leftarrow \rho_{(\text{Ssn, Net_Salary, Bonus})} (\pi_{\text{Ssn, Salary-Deduction, Years_service*1000}} (\text{EMPLOYEE_S}))$$

Единечна релациона операција RENAME

Пример

TEMP

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston,TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston,TX	M	40000	888665555	5
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble,TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

R

First_name	Last_name	Salary
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

Анализирајте ги релациите TEMP и R!

Кои се изразите во релациона алгебра со кои можете да ги добиете овие две релации ако работите со релациониот модел на COMPANY базата на податоци?

Операции од релационата алгебра од теоријата на множества UNION

$$R \cup S$$

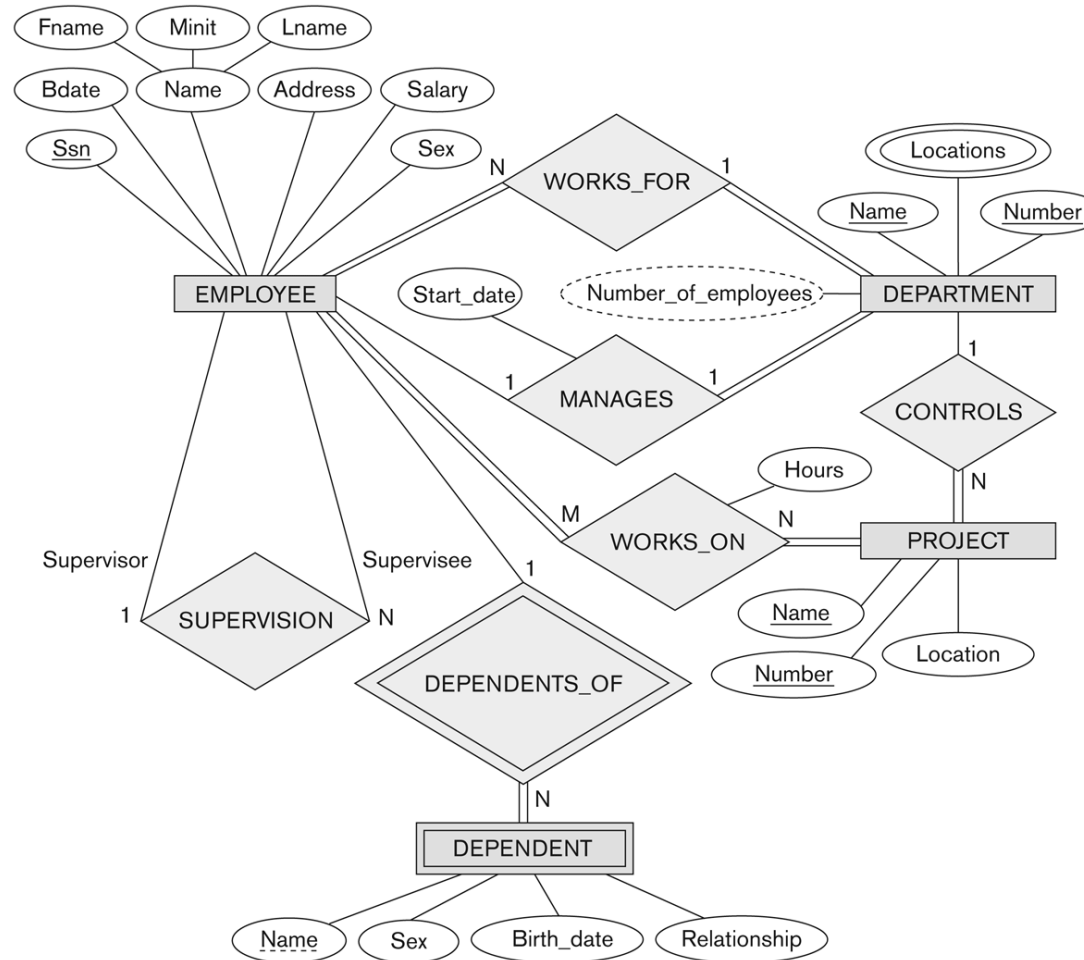
- Бинарна операција, се означува со \cup
- Резултатот од операцијата $R \cup S$, е релација која ги вклучува торките кои припаѓаат или на R или на S или во двете R и S
- Дупликат торките се отстрануваат
- Двете релации R и S мора да се “**компатибилни по тип**” (или UNION компатибилни)

Операции од релационата алгебра од теоријата на множества UNION

- За сите оператори од овој тип мора да постои компатибилност на операндите
- $R_1(A_1, A_2, \dots, A_n)$ и $R_2(B_1, B_2, \dots, B_n)$ се компатибилни по тип ако:
 - имаат ист број на атрибути (n), и
 - домените на соодветните атрибути се компатибилни по тип (може да се споредуваат)
$$\text{dom}(A_i) = \text{dom}(B_i) \text{ за } i = 1, 2, \dots, n$$
- Резултирачката релација ќе ги има истите имиња на атрибутите како првиот операнд R_1

Операции од релационата алгебра од теоријата на множества UNION

Пример



Да се пронајдат матичните броеви на сите вработени кои работат во оддел 5 или директно надгледуваат вработен кој работи во одделот 5?

EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)



Операции од релационата алгебра од теоријата на множества UNION



Пример

- Да се пронајдат матичните броеви на сите вработени кои работат во оддел 5 или директно надгледуваат вработен кој работи во одделот 5?
- Може да го искористиме операторот UNION на следниот начин:

$$\text{DEP5_EMPS} \leftarrow \sigma_{\text{Dno}=5} (\text{EMPLOYEE})$$
$$\text{RESULT1} \leftarrow \pi_{\text{ssn}} (\text{DEP5_EMPS})$$
$$\text{RESULT2} \leftarrow \pi_{\text{Super_ssn}} (\text{DEP5_EMPS})$$
$$\text{RESULT} \leftarrow \text{RESULT1} \cup \text{RESULT2}$$

Операции од релационата алгебра од теоријата на множества UNION

Пример

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEP5_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

$DEP5_EMPS \leftarrow \sigma_{Dno=5} (EMPLOYEE)$

$RESULT1 \leftarrow \pi_{Ssn}(DEP5_EMPS)$

$RESULT2 \leftarrow \pi_{Super_ssn}(DEP5_EMPS)$

$RESULT \leftarrow RESULT1 \cup RESULT2$

RESULT1

Ssn
123456789
333445555
666884444
453453453

RESULT

Ssn
123456789
333445555
666884444
453453453
888665555

RESULT2

Ssn
333445555
888665555



Операции од релационата алгебра од теоријата на множества INTERSECTION

$$R \cap S$$

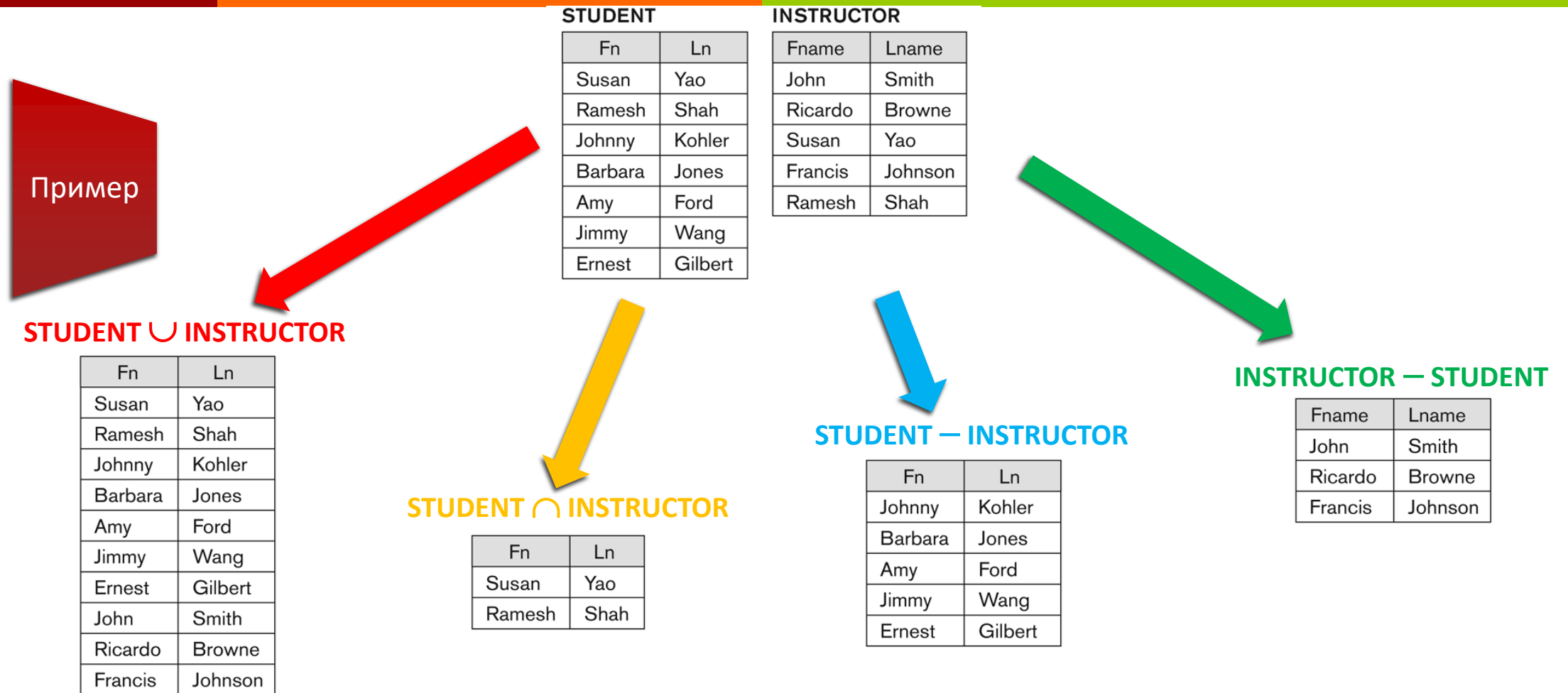
- INTERSECTION или пресек се означува со \cap
- Двата операнди, релациите R и S мора да се компатибилни по тип
- Резултатот од операцијата $R \cap S$, е релација која ги вклучува сите торки кои се заеднички за R и S
 - Имињата на атрибутите во резултантната релација ќе бидат исти како тие во релацијата R

Операции од релационата алгебра од теоријата на множества SET DIFFERENCE

$$R - S$$

- SET DIFFERENCE (исто се среќава и како MINUS или EXCEPT) е разлика на множества и се означува со –
- Резултатот од операцијата $R - S$, е релација која ги вклучува сите торки кои се среќаваат во R , но ги нема во S
 - Имињата на атрибутите во резултантната релација ќе бидат исти како тие во релацијата R
- Двата операнди, релациите R и S мора да се компатибилни по тип

Операции од релационата алгебра од теоријата на множества



Некои својства на UNION, INTERSECT и DIFFERENCE

- Унија и пресек се *комутативни* операции; односно важи:
 - $R \cup S = S \cup R$, и $R \cap S = S \cap R$
- И унијата и пресекот може да се бидат *n*-арни операции (да ги извршувате операциите над *n* релации наеднаш), а во исто време се и *асоцијативни* операции; односно важи:
 - $R \cup (S \cup T) = (R \cup S) \cup T$
 - $(R \cap S) \cap T = R \cap (S \cap T)$
- Операцијата разлика не е комутативна; односно важи:
 - $R - S \neq S - R$

Операции од релационата алгебра од теоријата на множества

ДЕКАРТОВ ПРОИЗВОД

$$R \times S$$

- Операцијата ДЕКАРТОВ (или CROSS, CARTESIAN) ПРОИЗВОД се користи за комбинирање на торките од две релации
 - Се означува со $R(A_1, A_2, \dots, A_n) \times S(B_1, B_2, \dots, B_m)$
 - Резултира во нова релација Q со степен од $n + m$ атрибути:
 - $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$, во овој редослед.
 - Состојбата на резултантната релација има по еден запис за секоја комбинација на торките - една од R и една од S .
 - Ако R имала n_R торки ($|R| = n_R$), и S имала n_S торки, тогаш резултатот $R \times S$ ќе има $n_R * n_S$ торки.
 - Двете релации **НЕ МОРА** да се компатибилни по тип

Операции од релационата алгебра од теоријата на множества ДЕКАРТОВ ПРОИЗВОД

R		X	S		=	Q			
A	B		B	C		A	B	B	C
1	2		2	6		1	2	2	6
1	4		2	4		1	2	2	4
2	5		3	5		1	2	3	5
			4	8		1	2	4	8
						1	4	2	6
						1	4	2	4
						1	4	3	5
						1	4	4	8
						2	5	2	6
						2	5	2	4
						2	5	3	5
						2	5	4	8



Операции од релационата алгебра од теоријата на множества

ДЕКАРТОВ ПРОИЗВОД

Пример

- Генерално, ДЕКАРТОВИОТ ПРОИЗВОД не резултира во семантички значаен резултат
 - Резултатот може да стане значаен доколку врз него се применат и други операции
- ПРИМЕР (не значајно):
 - $\text{FEMALE_EMPS} \leftarrow \sigma_{\text{Sex}='F'}(\text{EMPLOYEE})$
 - $\text{EMP_NAMES} \leftarrow \pi_{\text{Fname, Lname, Ssn}}(\text{FEMALE_EMPS})$
 - $\text{EMP_DEPENDENTS} \leftarrow \text{EMP_NAMES} \times \text{DEPENDENT}$
- EMP_DEPENDENTS ќе ја содржи секоја комбинација од EMP_NAMES и DEPENDENT без разлика дали тие се во некакво сродство или не се!

Операции од релационата алгебра од теоријата на множества

ДЕКАРТОВ ПРОИЗВОД

➤ ПРИМЕР (не значајно):

➤ $FEMALE_EMPS \leftarrow \sigma_{Sex='F'}(EMPLOYEE)$

➤ $EMP_NAMES \leftarrow \pi_{Fname, Lname, Ssn}(FEMALE_EMPS)$

➤ $EMP_DEPENDENTS \leftarrow EMP_NAMES \times DEPENDENT$

Пример

FEMALE_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Alicia	J	Zelaya	999887777	1968-07-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

EMP_NAMES

Fname	Lname	Ssn
Alicia	Zelaya	999887777
Jennifer	Wallace	987654321
Joyce	English	453453453

EMP_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Alicia	Zelaya	999887777	333445555	Alice	F	1986-04-05	...
Alicia	Zelaya	999887777	333445555	Theodore	M	1983-10-25	...
Alicia	Zelaya	999887777	333445555	Joy	F	1958-05-03	...
Alicia	Zelaya	999887777	987654321	Abner	M	1942-02-28	...
Alicia	Zelaya	999887777	123456789	Michael	M	1988-01-04	...
Alicia	Zelaya	999887777	123456789	Alice	F	1988-12-30	...
Alicia	Zelaya	999887777	123456789	Elizabeth	F	1967-05-05	...
Jennifer	Wallace	987654321	333445555	Alice	F	1986-04-05	...
Jennifer	Wallace	987654321	333445555	Theodore	M	1983-10-25	...
Jennifer	Wallace	987654321	333445555	Joy	F	1958-05-03	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...
Jennifer	Wallace	987654321	123456789	Michael	M	1988-01-04	...
Jennifer	Wallace	987654321	123456789	Alice	F	1988-12-30	...
Jennifer	Wallace	987654321	123456789	Elizabeth	F	1967-05-05	...
Joyce	English	453453453	333445555	Alice	F	1986-04-05	...
Joyce	English	453453453	333445555	Theodore	M	1983-10-25	...
Joyce	English	453453453	333445555	Joy	F	1958-05-03	...
Joyce	English	453453453	987654321	Abner	M	1942-02-28	...
Joyce	English	453453453	123456789	Michael	M	1988-01-04	...
Joyce	English	453453453	123456789	Alice	F	1988-12-30	...
Joyce	English	453453453	123456789	Elizabeth	F	1967-05-05	...

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse



Операции од релационата алгебра од теоријата на множества ДЕКАРТОВ ПРОИЗВОД

Пример

➤ ПРИМЕР (значајно):

➤ $FEMALE_EMPS \leftarrow \sigma_{Sex='F'}(EMPLOYEE)$

➤ $EMP_NAMES \leftarrow \pi_{Fname, Lname, Ssn}(FEMALE_EMPS)$

➤ $EMP_DEPENDENTS \leftarrow EMP_NAMES \times DEPENDENT$

➤ $ACTUAL_DEPS \leftarrow \sigma_{Ssn=Essn}(EMP_DEPENDENTS)$

➤ $RESULT \leftarrow \pi_{Fname, Lname, Dependent_name}(ACTUAL_DEPS)$

➤ RESULT ќе ги содржи имињата на вработените од женски пол и нивните издржувани лица

EMP_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Alicia	Zelaya	999887777	333445555	Alice	F	1986-04-05	...
Alicia	Zelaya	999887777	333445555	Theodore	M	1983-10-25	...
Alicia	Zelaya	999887777	333445555	Joy	F	1958-05-03	...
Alicia	Zelaya	999887777	987654321	Abner	M	1942-02-28	...
Alicia	Zelaya	999887777	123456789	Michael	M	1988-01-04	...
Alicia	Zelaya	999887777	123456789	Alice	F	1988-12-30	...
Alicia	Zelaya	999887777	123456789	Elizabeth	F	1967-05-05	...
Jennifer	Wallace	987654321	333445555	Alice	F	1986-04-05	...
Jennifer	Wallace	987654321	333445555	Theodore	M	1983-10-25	...
Jennifer	Wallace	987654321	333445555	Joy	F	1958-05-03	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...
Jennifer	Wallace	987654321	123456789	Michael	M	1988-01-04	...
Jennifer	Wallace	987654321	123456789	Alice	F	1988-12-30	...
Jennifer	Wallace	987654321	123456789	Elizabeth	F	1967-05-05	...
Joyce	English	453453453	333445555	Alice	F	1986-04-05	...
Joyce	English	453453453	333445555	Theodore	M	1983-10-25	...
Joyce	English	453453453	333445555	Joy	F	1958-05-03	...
Joyce	English	453453453	987654321	Abner	M	1942-02-28	...
Joyce	English	453453453	123456789	Michael	M	1988-01-04	...
Joyce	English	453453453	123456789	Alice	F	1988-12-30	...
Joyce	English	453453453	123456789	Elizabeth	F	1967-05-05	...

ACTUAL_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...

RESULT

Fname	Lname	Dependent_name
Jennifer	Wallace	Abner



Операции од релационата алгебра од теоријата на множества

ДЕКАРТОВ ПРОИЗВОД

Пример

➤ ПРИМЕР (значајно):

➤ $\text{FEMALE_EMPS} \leftarrow \sigma_{\text{Sex}='F'}(\text{EMPLOYEE})$

➤ $\text{EMP_NAMES} \leftarrow \pi_{\text{Fname, Lname, Ssn}}(\text{FEMALE_EMPS})$

➤ $\text{EMP_DEPENDENTS} \leftarrow \text{EMP_NAMES} \times \text{DEPENDENT}$

➤ $\text{ACTUAL_DEPS} \leftarrow \sigma_{\text{Ssn=Essn}}(\text{EMP_DEPENDENTS})$

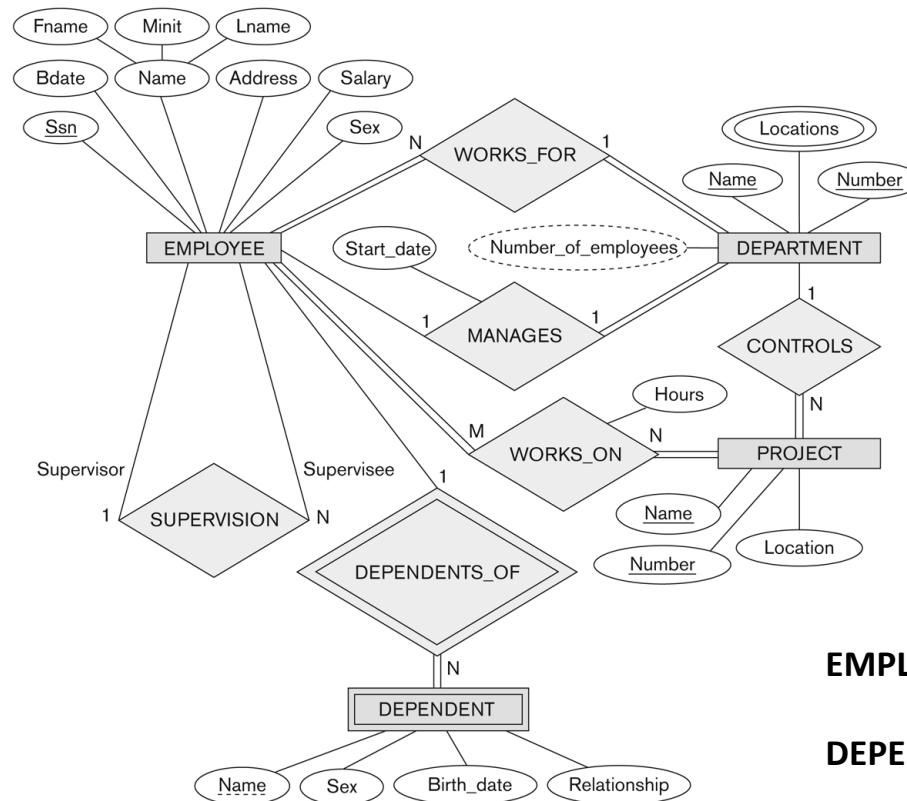
➤ $\text{RESULT} \leftarrow \pi_{\text{Fname, Lname, Dependent_name}}(\text{ACTUAL_DEPS})$

➤ RESULT ќе ги содржи имињата на вработените од женски пол и нивните издржувани лица

ЗОШТО?

Операции од релационата алгебра од теоријата на множества ДЕКАРТОВ ПРОИЗВОД

Пример



ПРИМЕР (значајно):

➤ **FEMALE_EMPS** $\leftarrow \sigma_{\text{Sex}='F'}(\text{EMPLOYEE})$

➤ **EMPNAMES** $\leftarrow \pi_{\text{Fname, Lname, Ssn}}(\text{FEMALE_EMPS})$

➤ **EMP_DEPENDENTS** $\leftarrow \text{EMPNAMES} \times \text{DEPENDENT}$

➤ **ACTUAL_DEPS** $\leftarrow \sigma_{\text{Ssn}=\text{Essn}}(\text{EMP_DEPENDENTS})$

➤ **RESULT** $\leftarrow \pi_{\text{Fname, Lname, Dependent_name}}(\text{ACTUAL_DEPS})$



RESULT ќе ги содржи имињата на вработените од женски пол и нивните издржувани лица

EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

DEPENDENT(Essn*,Dependent_name,Sex,Bdate,Relationship)



Бинарни релациони операции : JOIN

$$R \bowtie_{\langle \text{join condition} \rangle} S$$

- Операцијата на спојување JOIN (се означува со \bowtie) ја комбинира во единствена операција секвенцата од ДЕКАРТОВ ПРОИЗВОД следена од SELECT
 - Секвенцата од ДЕКАРТОВ ПРОИЗВОД следена од SELECT често се користи за идентификација и селекција на **поврзаните торки** од две релации (комбинациите кои имаат семантичка смисла)
 - Оваа операција е многу важна за секоја релациона БП во која има повеќе релации, затоа што овозможува *комбинирање на поврзаните торки* од различните релации
 - Релациите R и S може да бидат било какви релации кои резултираат од општите *изрази од релационата алгебра*.

Бинарни релациони операции : JOIN

R		$\bowtie_{R.B=S.B}$	S		=	RESULT			
A	B		B	C		A	B	B	C
1	2		2	6		1	2	2	6
1	4		2	4		1	2	2	4
2	5		3	5					
			4	8		1	4	4	8

Бинарни релациони операции : JOIN

$$R \bowtie_{R.B=S.B} S = \sigma_{R.B=S.B} (R \times S)$$

$$\sigma_{R.B=S.B} \left[\begin{array}{c|c} \mathbf{R} & \mathbf{S} \\ \hline \begin{array}{cc} A & B \\ 1 & 2 \\ 1 & 4 \\ 2 & 5 \end{array} & \begin{array}{cc} B & C \\ 2 & 6 \\ 2 & 4 \\ 3 & 5 \\ 4 & 8 \end{array} \end{array} \right] \times = \sigma_{R.B=S.B}$$

INTERMEDIATE			
A	R.B	S.B	C
1	2	2	6
1	2	2	4
1	2	3	5
1	2	4	8
1	4	2	6
1	4	2	4
1	4	3	5
1	4	4	8
2	5	2	6
2	5	2	4
2	5	3	5
2	5	4	8

=

RESULT			
A	B	B	C
1	2	2	6
1	2	2	4
1	4	4	8



Бинарни релациони операции : JOIN

- Генералниот случај на JOIN операцијата се нарекува Theta-join:



- Условот на спојување се нарекува *theta* или Θ
- Θ може да биде било кој логички израз од атрибутите на R и S
 - На пример: $R.A_i < S.B_j \text{ AND } (R.A_k = S.B_l \text{ OR } R.A_p < S.B_q)$
- Повеќето услови за спојување вклучуваат еден или повеќе услови на изедначување споени со AND логичкиот оператор
 - На пример: $R.A_i = S.B_j \text{ AND } R.A_k = S.B_l \text{ AND } R.A_p = S.B_q$

Некои карактеристики на JOIN

➤ Нека ја разгледаме следната JOIN операција:

$$\begin{array}{c} \text{➤ } R(A_1, A_2, \dots, A_n) \quad \bowtie \quad S(B_1, B_2, \dots, B_m) \\ R.A_i = S.B_j \end{array}$$

➤ Резултатот ќе биде некоја релација Q со степен од $n + m$ атрибути:

➤ $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$, по овој редослед

➤ Состојбата на резултантната релација ќе има по една торка за секоја комбинација од торките - r од R и s од S , но *само ако тие го задоволуваат условот за спојување $r[A_i] = s[B_j]$*

➤ Ако R има n_R торки, и S има n_S торки, тогаш во резултатот од спојувањето во општ случај ќе има *најмногу* $n_R * n_S$ торки

Бинарни релациони операции : EQUIJOIN

- Најчестото користење на join операторот вклучува услови за спојување со *споредба на еднаквоста*
 - Најголемо семантичко значење има изедначување на примарен со нему соодветен надворешен клуч (ова е најчестото изедначување кај релационите бази на податоци)
- Ваквиот тип на join, каде се користи само операторот =, се нарекува **EQUIJOIN**
- Во резултатот на EQUIJOIN секогаш имаме еден или повеќе парови од атрибути (чии имиња не мора да се исти) кои имаат идентични вредности во секоја од торките.

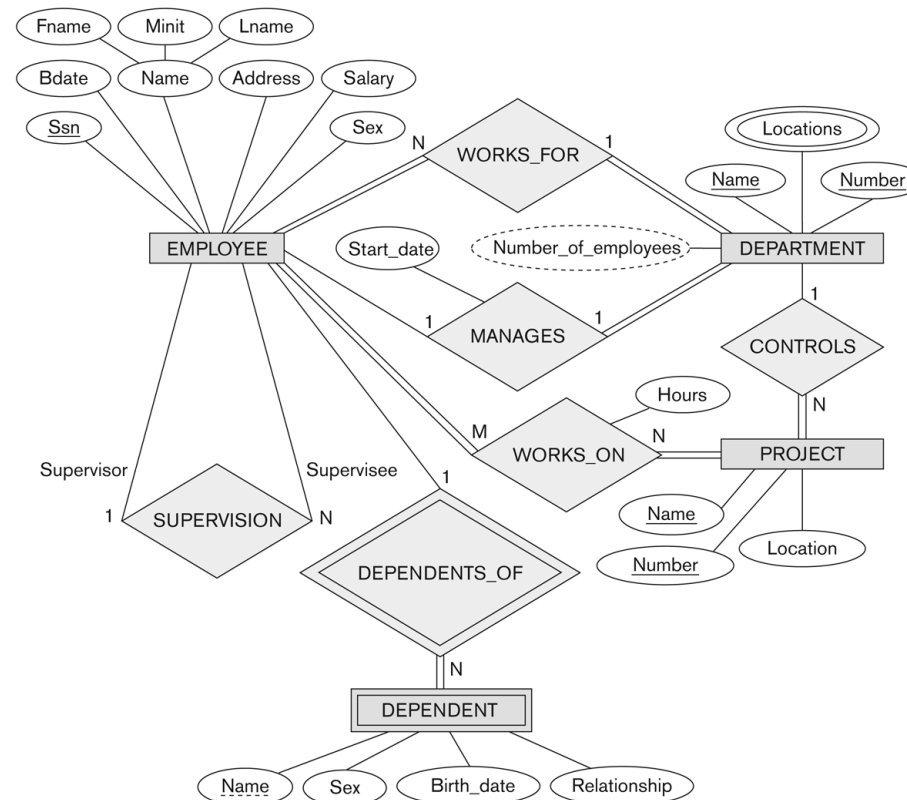
Бинарни релациони операции : EQUIJOIN

Пример

➤ Да се пронајдат имињата на управниците на секој оддел?

➤ EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

➤ DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)



Бинарни релациони операции : EQUIJOIN

Пример

- Да се пронајдат имињата на управниците на секој оддел?
 - EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)
 - DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)
 - За да се добијат имињата на управниците, треба да се искомбинираат секоја DEPARTMENT торка со EMPLOYEE торката чија Ssn вредност се поклопува со Mgr_ssn вредноста сместена во торката на одделот.

DEPT_MGR ← DEPARTMENT  EMPLOYEE
Mgr_ssn=Ssn

- Mgr_ssn=Ssn е условот за спојување
 - Ги комбинира сите записи од одделите со соодветните вработени кои управуваат со истите
 - Условот за спојување може да се напише и во следниот формат
DEPARTMENT.Mgr_ssn= EMPLOYEE.Ssn

Бинарни релациони операции : EQUIJOIN

Пример

DEPT_MGR

Dname	Dnumber	Mgr_ssn	...	Fname	Minit	Lname	Ssn	...
Research	5	333445555	...	Franklin	T	Wong	333445555	...
Administration	4	987654321	...	Jennifer	S	Wallace	987654321	...
Headquarters	1	888665555	...	James	E	Borg	888665555	...

DEPT_MGR ← DEPARTMENT  EMPLOYEE
Mgr_ssn=Ssn

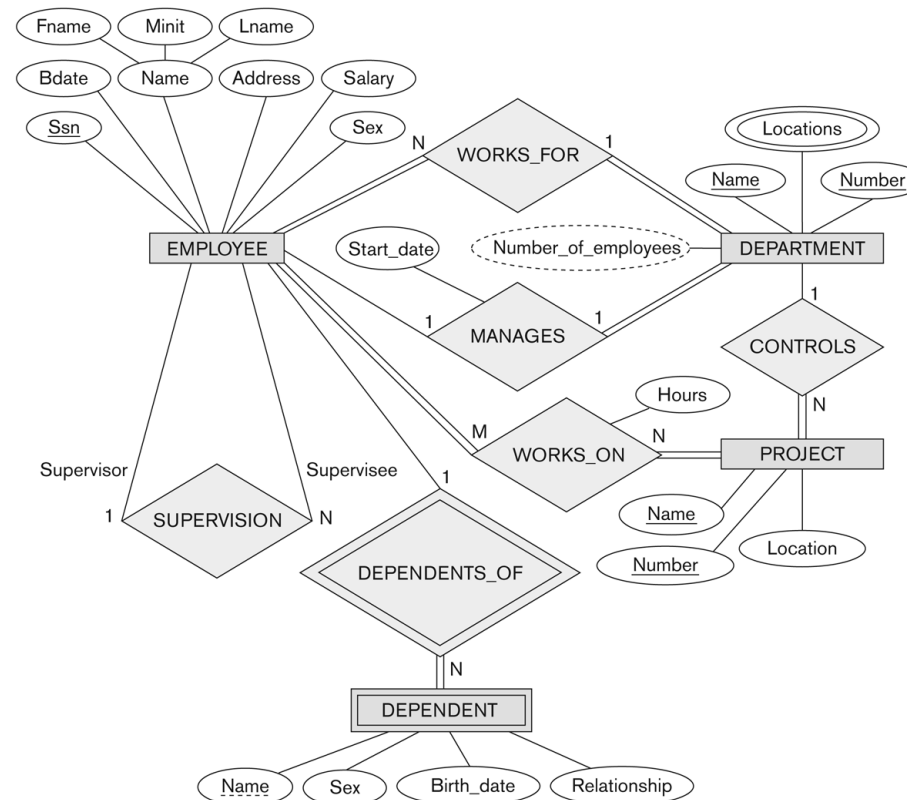
Бинарни релациони операции : NATURAL JOIN

- Друга варијанта на JOIN оперторот, наречена **NATURAL JOIN** — се означува со ***** — е креирана за да се справи со вториот атрибут (дупликатот) кој се добива од EQUIJOIN условот.
- Стандардната дефиниција на природното соединување бара двата атрибути за спојување, или секој пар атрибути за спојување, да **имаат исто име** во двете релации
 - Ако ова не е случај, претходно мора да се вклучи операцијата на преименување (RENAME).

Бинарни релациони операции : NATURAL JOIN

Пример

- Да се добијат одделите со нивните локации?
- DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)
- DEPT_LOCATIONS(Dnumber*,Dlocations)



Бинарни релациони операции : NATURAL JOIN

Пример

➤ Да се добијат одделите со нивните локации?

➤ DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)

➤ DEPT_LOCATIONS(Dnumber*,Dlocation)

DEPT_LOCS ← DEPARTMENT * DEPT_LOCATIONS

➤ Имплицитен услов за спојување се креира врз основа на атрибутите по кои ќе се спојува: DEPARTMENT.Dnumber = DEPT_LOCATIONS.Dnumber

DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

DEPT_LOCS

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date	Location
Headquarters	1	888665555	1981-06-19	Houston
Administration	4	987654321	1995-01-01	Stafford
Research	5	333445555	1988-05-22	Bellaire
Research	5	333445555	1988-05-22	Sugarland
Research	5	333445555	1988-05-22	Houston



Бинарни релациони операции : NATURAL JOIN

- Во општ случај ако релациите имаат повеќе заеднички атрибути

$$Q \leftarrow R(A,B,C,D) * S(C,D,E)$$

- Имплицитен услов за спојување го вклучува **секој пар** од атрибутите кои имаат исто име поврзани со AND операторот:

- $R.C = S.C \text{ AND } R.D = S.D$

- Во резултатот ќе се смести само еден од атрибутите од секој пар:

- $Q(A,B,C,D,E)$

Бинарни релациони операции : DIVISION

$$R(Z) \div S(X)$$

- $R(Z) \div S(X)$, каде $X \subseteq Z$. Нека $Y = Z - X$ (и според тоа $Z = X \cup Y$); односно Y ќе биде множество на атрибути од R кои не се атрибути на S .
- Резултатот од операцијата DIVISION е релација $T(Y)$ која вклучува торка t ако торките t_R се појавуваат во R со $t_R[Y] = t$, и со $t_R[X] = t_s$ за секоја торка t_s во S .
- За торката t да се појави во резултатот од DIVISION, вредностите во t мора да се појавуваат во R во комбинација со секоја торка во S .

Бинарни релациони операции : DIVISION

Пример

Резултат од извршувањето на:

$$T = R \div S$$

- По дефиниција $R(Z) \div S(X)$
 - $Z=\{A,B\}$, $X=\{A\}$, важи $X \subseteq Z$
- $Y = Z - X = \{A,B\} - \{A\} = \{B\}$
- Прашањето е: Дали постојат вредности на атрибутот B во табелата R кои се појавуваат во комбинација со секоја можна вредност на атрибутот A од табелата S?

R	
A	B
a1	b1
a2	b1
a3	b1
a4	b1
a1	b2
a3	b2
a2	b3
a3	b3
a4	b3
a1	b4
a2	b4
a3	b4

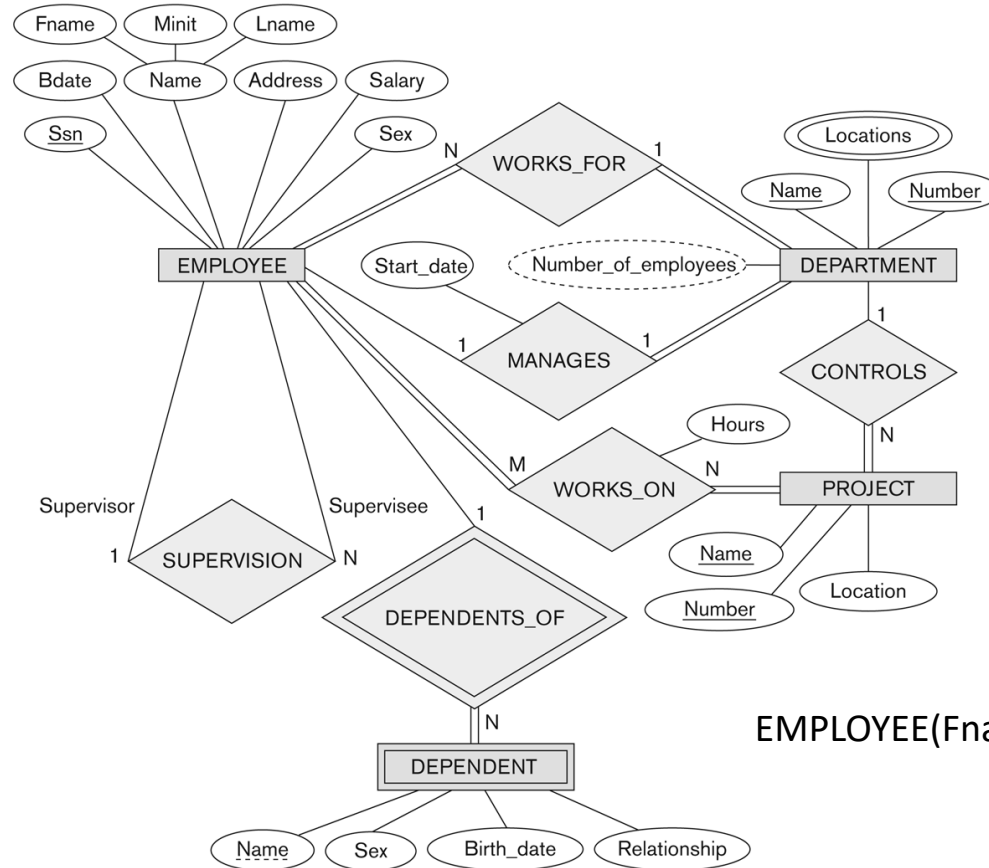
S
A
a1
a2
a3

T
B
b1
b4

Бинарни релациони операции : DIVISION

➤ Дај ги имињата на вработените кои работат на истите проекти на кои работи и 'John Smith'

Пример



DEPARTMENT(Dname,Dnumber,Mgr_ssn*,Mgr_start_date)

DEPT_LOCATIONS(Dnumber*,Dlocations)

PROJECT(Pname,Pnumber,Plocation,Dnum*)

WORKS_ON(Essn*,Pno*,Hours)

DEPENDENT(Essn*,Dependent_name,Sex,Bdate,Relationship)

EMPLOYEE(Fname,Minit,Lname,Ssn,Bdate,Address,Sex,Salary,Super_ssn*,Dno*)

Бинарни релациони операции : DIVISION

Пример

➤ Дај ги имињата на вработените кои работат на истите проекти на кои работи и 'John Smith'

➤ $SMITH \leftarrow \sigma_{Fname='John' \text{ AND } Lname='Smith'}(EMPLOYEE)$

➤ $SMITH_PNOS \leftarrow \pi_{Pno} (WORKS_ON \bowtie_{Essn=ssn} SMITH)$

➤ $SSN_PNOS \leftarrow \pi_{Essn, Pno} (WORKS_ON)$

➤ $SSNS \leftarrow SSN_PNOS \div SMITH_PNOS$

SSN_PNOS

Essn	Pno
123456789	1
123456789	2
666884444	3
453453453	1
453453453	2
333445555	2
333445555	3
333445555	10
333445555	20
999887777	30
999887777	10
987987987	10
987987987	30
987654321	30
987654321	20
888665555	20

SMITH_PNOS

Pno
1
2

SSNS

Ssn
123456789
453453453

Комплетно множество на релациони операции

➤ Множеството на операциите SELECT σ , PROJECT π , UNION \cup , DIFFERENCE $-$, RENAME ρ и ДЕКАРТОВ ПРОИЗВОД \times се нарекува и **комплетно множество** бидејќи останатите оператори од релационата алгебра може да се изведат врз основа на овие оператори.

➤ На пример:

$$\rightarrow R \cap S = (R \cup S) - ((R - S) \cup (S - R))$$

$$\rightarrow R \bowtie_{\langle \text{join condition} \rangle} S = \sigma_{\langle \text{join condition} \rangle} (R \times S)$$

Преглед на операциите од релационата алгебра

Table 6.1
Operations of Relational Algebra

Operation	Purpose	Notation
SELECT	Selects all tuples that satisfy the selection condition from a relation R .	$\sigma_{\langle \text{selection condition} \rangle}(R)$
PROJECT	Produces a new relation with only some of the attributes of R , and removes duplicate tuples.	$\pi_{\langle \text{attribute list} \rangle}(R)$
THETA JOIN	Produces all combinations of tuples from R_1 and R_2 that satisfy the join condition.	$R_1 \bowtie_{\langle \text{join condition} \rangle} R_2$
EQUIJOIN	Produces all the combinations of tuples from R_1 and R_2 that satisfy a join condition with only equality comparisons.	$R_1 \bowtie_{\langle \text{join condition} \rangle} R_2$, OR $R_1 \bowtie_{(\langle \text{join attributes 1} \rangle), (\langle \text{join attributes 2} \rangle)} R_2$
NATURAL JOIN	Same as EQUIJOIN except that the join attributes of R_2 are not included in the resulting relation; if the join attributes have the same names, they do not have to be specified at all.	$R_1 *_{\langle \text{join condition} \rangle} R_2$, OR $R_1 *_{(\langle \text{join attributes 1} \rangle), (\langle \text{join attributes 2} \rangle)} R_2$ OR $R_1 * R_2$
UNION	Produces a relation that includes all the tuples in R_1 or R_2 or both R_1 and R_2 ; R_1 and R_2 must be union compatible.	$R_1 \cup R_2$
INTERSECTION	Produces a relation that includes all the tuples in both R_1 and R_2 ; R_1 and R_2 must be union compatible.	$R_1 \cap R_2$
DIFFERENCE	Produces a relation that includes all the tuples in R_1 that are not in R_2 ; R_1 and R_2 must be union compatible.	$R_1 - R_2$
CARTESIAN PRODUCT	Produces a relation that has the attributes of R_1 and R_2 and includes as tuples all possible combinations of tuples from R_1 and R_2 .	$R_1 \times R_2$
DIVISION	Produces a relation $R(X)$ that includes all tuples $t[X]$ in $R_1(Z)$ that appear in R_1 in combination with every tuple from $R_2(Y)$, where $Z = X \cup Y$.	$R_1(Z) \div R_2(Y)$



Нотација за дрва од прашалници

- Дрво од прашалници (Query Tree)
 - Внатрешна податочна структура за репрезентација на прашалниците
 - Во внатрешните јазли се сместуваат операциите како што се селекција, проекција, спојување, преименување,
 - Во листовите се сместуваат основните релации над кои ќе се изврши прашалникот
 - Дрвото дава добра визуелна претстава на комплексноста на прашалникот и на употребените операции
 - Стандардна техника за оптимизација на извршувањето на прашалниците

Нотација за дрва од прашалници

кој е изразот од релационата алгебра претставен со ова дрво на прашалници?

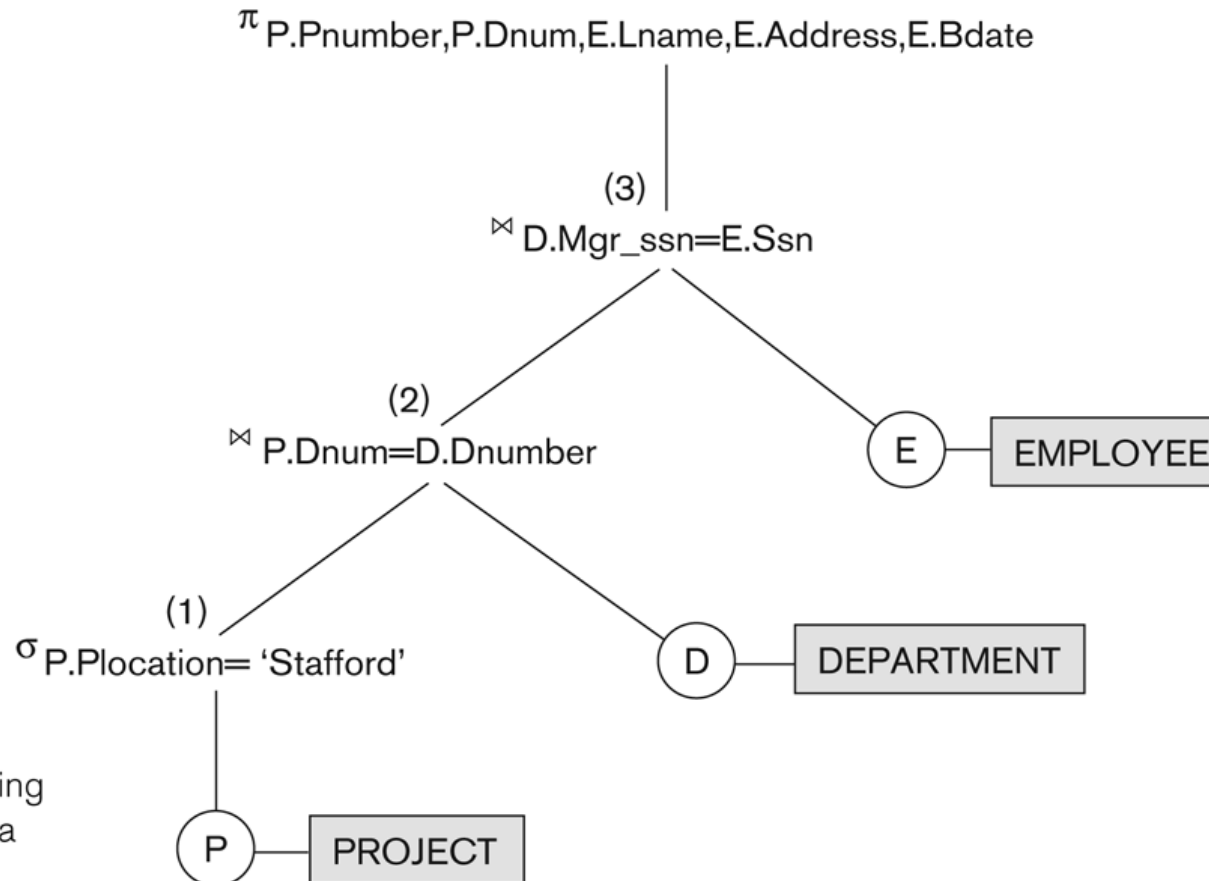


Figure 6.9

Query tree corresponding to the relational algebra expression for Q2.

Дополнителни релациони операции

➤ Како дополнителни релациони функции ќе ги запознаеме:

- агрегатните функции
- функциите за групирање
- операцијата OUTER JOIN
- операцијата OUTER UNION

Агрегатни функции

- Типот на барање кој не може да се изрази преку основните операции на релационата алгебра е спецификацијата на математичката **агрегатна функција** врз колекции на вредности од базата на податоци.
- Пример функции кои се користат во прости статистички прашалници кои ја сумаризираат информацијата од торките во БП:
 - пресметка на просечна или вкупна плата на сите вработени или
 - пребројување на бројот на торки во некоја од релациите.
- Најчестите функции применувани над колекција од нумерички вредности вклучуваат пресметка на
 - SUM, AVERAGE, MAXIMUM, и MINIMUM.
- Функцијата COUNT се користи за броење на торки или вредности.

Операција за агрегација

\mathcal{F} агрегатна функција (R)

- Се користи операцијата за агрегација \mathcal{F} и соодветната агрегатна функција:
 - $\mathcal{F}_{\text{MAX Salary}}$ (EMPLOYEE) максимална плата во релацијата EMPLOYEE
 - $\mathcal{F}_{\text{MIN Salary}}$ (EMPLOYEE) минимална плата во релацијата EMPLOYEE
 - $\mathcal{F}_{\text{SUM Salary}}$ (EMPLOYEE) вкупна плата на сите вработени
 - $\mathcal{F}_{\text{COUNT Ssn, AVERAGE Salary}}$ (EMPLOYEE) пресметува број на вработени и нивната просечна плата
 - Агрегатните функции не отстрануваат дупликат вредности од атрибутот врз кој се применуваат
 - Агрегатните функции не враќаат скалар (вредност) туку враќаат релација која има една единствена торка

Групирање во агрегацијата

групирачки атрибут \mathcal{F} агрегатна функција (R)

- Сите претходни примери сумаризираа еден или повеќе атрибути над множество од торки (максимална плата или број на вработени)
- Преку групирањето се одвојуваат подмножества од множеството на торки врз основа на вредностите кои може да ги прими некој атрибут (овој атрибут се нарекува групирачки атрибут)
- На пример: $\text{Dno } \mathcal{F}_{\text{COUNT Ssn, AVERAGE Salary}} (\text{EMPLOYEE})$

Агрегатни функции и групирање

Пример

$\mathcal{F}_{\text{COUNT Ssn, AVERAGE Salary}}(\text{EMPLOYEE})$

Count_ssn	Average_salary
8	35125

$\text{DNO } \mathcal{F}_{\text{COUNT Ssn, AVERAGE Salary}}(\text{EMPLOYEE})$

Fname	Minit	Lname	<u>Ssn</u>	...	Salary	Super_ssn	Dno
John	B	Smith	123456789		30000	333445555	5
Franklin	T	Wong	333445555		40000	888665555	5
Ramesh	K	Narayan	666884444		38000	333445555	5
Joyce	A	English	453453453	...	25000	333445555	5
Alicia	J	Zelaya	999887777		25000	987654321	4
Jennifer	S	Wallace	987654321		43000	888665555	4
Ahmad	V	Jabbar	987987987		25000	987654321	4
James	E	Bong	888665555		55000	NULL	1

Како да се променат
имињата на овие
атрибути?

Dno	Count (*)	Avg (Salary)
5	4	33250
4	3	31000
1	1	55000

Дополнителни релациони операции

➤ OUTER JOIN операција

- Во NATURAL JOIN и EQUIJOIN, торките кои *не се спојуваат* (не се во некој однос) се елиминираат од резултатот на спојувањето
 - Торките со **null вредност** во join атрибутите се елиминираат
 - Може да се појави губење на информации
- Множеството на операции, наречени OUTER joins, може да се користи кога сакаме да ги задржиме
 - сите торки од R, или
 - сите торки од S, или
 - сите торки од двете релации во резултатот од спојувањето,без разлика дали тие имаат одговарачки торки во другата релација или не

Дополнителни релациони операции

- **Левиот outer join** ги задржува сите торки од првата релација R од операцијата R S; ако нема соодветна торка од другата релација S, тогаш атрибутите од S во резултатот се пополнуваат со null вредности \bowtie
- Обратно на левото спојување, имаме **десен outer join**, кој ги чува сите торки од втората или десната релација S во операцијата спојување $R \bowtie S$.
- Третата операција, **целосен outer join**, означен со \bowtie ги чува сите торки и од левата и од десната релација, при што ако нема спојливи торки, тогаш се користат null вредности на местата каде е потребно

Дополнителни релациони операции

Пример

TEMP \leftarrow EMPLOYEE \bowtie DEPARTMENT

Ssn = Mgr_ssn

RESULT $\leftarrow \pi_{\text{Fname, Minit, Lname, Dname}}(\text{TEMP})$

RESULT

Fname	Minit	Lname	Dname
John	B	Smith	NULL
Franklin	T	Wong	Research
Alicia	J	Zelaya	NULL
Jennifer	S	Wallace	Administration
Ramesh	K	Narayan	NULL
Joyce	A	English	NULL
Ahmad	V	Jabbar	NULL
James	E	Borg	Headquarters

Врати ги имињата на вработените заедно со имињата на одделите на кои управуваат (доколку вработениот е управник)

NULL вредности во релациона алгебра

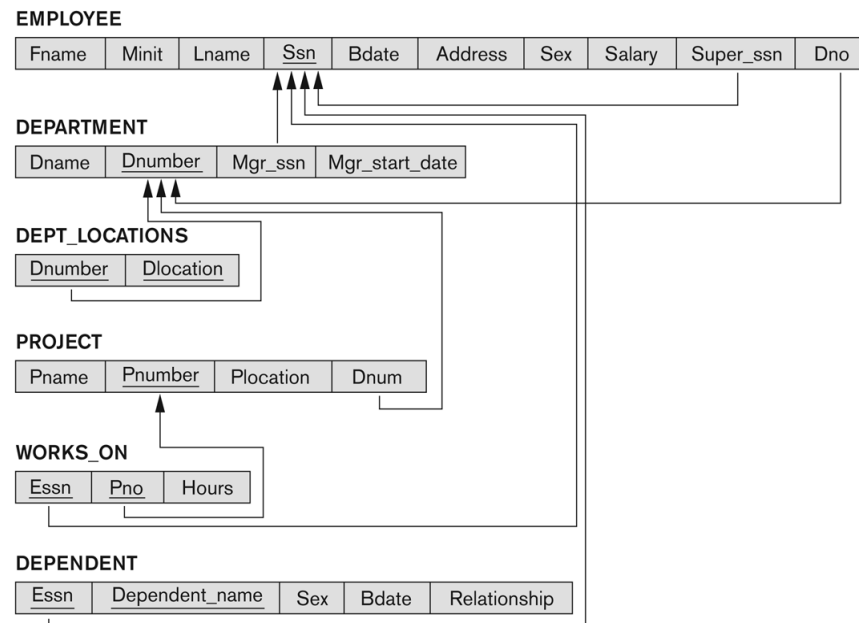
- NULL вредноста означува непозната или непостоечка вредност
- Резултатот на било која аритметичка операција со единствен аргумент NULL е повторно NULL
- Агрегатните функции ги игнорираат NULL вредностите
- Кога станува збор за елиминација на дупликати и групирање NULL се третира како било која друга вредност
 - Ова значи дека може да постои групирање во кое една или повеќе вредности се NULL
 - Две NULL вредности се сметаат за идентични (иако тоа семантички не секогаш е оправдано)
- Торките каде атрибутот по кој се извршува операцијата поврзување (join) е NULL не влегуваат во резултатот од поврзувањето
 - Не може да се направи Join по NULL вредности

Примери на прашања во релационата алгебра

- Q1: Најди ги имињата и адресите на сите вработени кои работат во 'Research' одделот.

$$\text{RESEARCH_DEPT} \leftarrow \sigma_{\text{Dname}='Research'}(\text{DEPARTMENT})$$

$$\text{RESEARCH_EMPS} \leftarrow (\text{RESEARCH_DEPT} \bowtie_{\text{Dnumber} = \text{Dno}} \text{EMPLOYEE})$$

$$\text{RESULT} \leftarrow \pi_{\text{Fname, Lname, Address}}(\text{RESEARCH_EMPS})$$


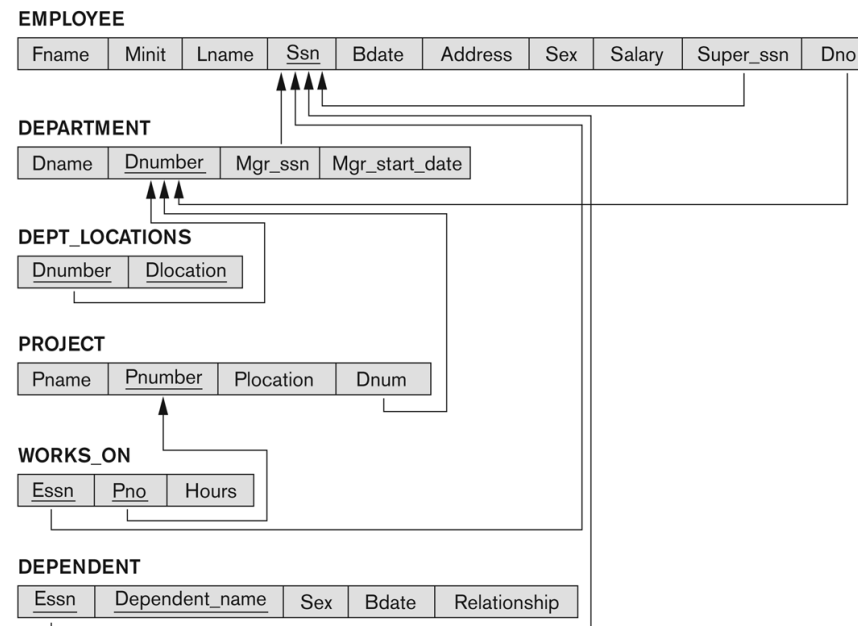
Примери на прашања во релационата алгебра

- Q6: Најди ги имињата на вработените кои немаат издржувани лица.

$$\text{ALL_EMPS} \leftarrow \pi_{\text{Ssn}}(\text{EMPLOYEE})$$

$$\text{EMPS_WITH_DEPS}(\text{Ssn}) \leftarrow \pi_{\text{Essn}}(\text{DEPENDENT})$$

$$\text{EMPS_WITHOUT_DEPS} \leftarrow (\text{ALL_EMPS} - \text{EMPS_WITH_DEPS})$$

$$\text{RESULT} \leftarrow \pi_{\text{Lname, Fname}}(\text{EMPS_WITHOUT_DEPS} * \text{EMPLOYEE})$$


Примери на прашања во релационата алгебра (2)

- Истите прашања, ама сега напишани со единечни изрази
- **Q1: Најди ги имињата и адресите на сите вработени кои работат во 'Research' одделот.**

$\pi_{Fname, Lname, Address} (\sigma_{Dname = 'Research'} (DEPARTMENT \bowtie_{Dnumber=Dno} EMPLOYEE))$

- **Q6: Најди ги имињата на вработените кои немаат издржувани лица.**

$\pi_{Lname, Fname} ((\pi_{Ssn} (EMPLOYEE) - \rho_{Ssn} (\pi_{Essn} (DEPENDENT))) * EMPLOYEE)$

Изборот кој начин ќе го користите е индивидуален.

Најважно е да знаете да ги трансформирате едниот тип изрази во другиот, и обратно!

Релациона пресметка (Relational Calculus)

- Изразот од **релационата пресметка** креира нова релација, која се специфицира преку **вредностите на променливите** кои се чуваат во редиците од релациите од БП (во **релационата пресметка над торка**) или врз колоните од соодветните релации на БП (во **релационата пресметка над домен**).
- Овде не постои подреденост во спецификацијата на тоа како да се добие резултатот од прашалникот
 - изразот од релационата пресметка специфицира која информација треба да се содржи во резултатот
 - ова е основната разлика помеѓу релационата алгебра и релационата пресметка

Релациона пресметка (Relational Calculus)

- Релационата пресметка може да се третира како **не-процедурален** или **декларативен** јазик
- И ова ја разликува од релационата алгебра, каде мора да напишеме секвенца од операции за да се специфицира прашалникот; според ова релационата алгебра може да се смета како **процедурален** начин на генерирање на прашалникот

Релациона пресметка врз торка

- Релационата пресметка врз торка се базира на специфицирањето на бројот на променливи на торки
- Секоја променлива на торка најчесто варира од релација до релација во БП, што значи дека променливата може да прими вредност од било која индивидуална торка од релацијата
- Едноставната релациона пресметка над торка е од следната форма:

$$\{t \mid \text{COND}(t)\}$$

- каде t е торка променливата и $\text{COND}(t)$ е условен израз која ја вклучува торката t .
- Резултатот од ова прашање ќе биде множество на сите торки t кои го исполнуваат условот $\text{COND}(t)$.

Релациона пресметка врз торка

- Најди ги имињата и презимињата на вработените кои имаат плата над \$50,000?

{t.FNAME, t.LNAME | EMPLOYEE(t) AND t.SALARY>50000}

- Условот EMPLOYEE(t) специфицира дека **опсегот на релацијата** од торка променливата t е EMPLOYEE.
- Во резултатот ќе се врати името и презимето од секоја EMPLOYEE торка t која го исполнува условот t.SALARY>50000

Егзистенцијален и Универзален Квантификатор

- Два специјални симболи наречени квантификатори може да се појават во формулите; ова се **универзалниот квантификатор** (\forall) и **егзистенцијалниот квантификатор** (\exists).
- Торка променливата t се врзува кога се квантифицира, односно се јавува во ($\forall t$) или ($\exists t$) изразот; инаку, торката е слободна.
- Ако F е формула, тогаш ($\exists t$)(F) и ($\forall t$)(F), каде t е торка променлива
 - Формулата ($\exists t$)(F) е точна ако формулата F евалуира точно за некои (најмалку една) торки слободно придружени појавувања на t во F ; инаку ($\exists t$)(F) ќе биде неточна.
 - Формулата ($\forall t$)(F) е точна ако формулата F евалуира во точно за секоја торка слободно придружени појавувања на t во F ; инаку ($\forall t$)(F) ќе биде грешна.

Егзистенцијален и Универзален Квантификатор

- \forall се нарекува универзален или “за сите” квантификатор затоа што секоја торка од универзумот на торки мора да ја евалуира формулата F во точно (true).
- \exists се нарекува егзистенцијален или “постои” квантификатор затоа што некоја торка која постои во универзумот на торки може да ја евалуира формулата F во точно (true).

Егзистенцијален и Универзален Квантификатор

- Најди ги имињата и адресите на сите вработени кои работат во 'Research' одделот.

{t.FNAME, t.LNAME, t.ADDRESS | EMPLOYEE(t) and (∃ d)
(DEPARTMENT(d) and **d.DNAME='Research'** and **d.DNUMBER=t.DNO**) }

- Единствените слободни променливи на торка во изразите од релационата пресметка треба да се појават лево од вертикалната линија (|).
 - Во претходниот израз, t е единствената слободна променлива; таа *успешно се врзува* со секоја торка.
- Ако торката го исполнува условот специфициран во прашалникот, атрибутите FNAME, LNAME и ADDRESS се враќаат за секоја трока.
 - Условите EMPLOYEE (t) и DEPARTMENT(d) ги специфицираат опсезите за t и d.
 - Условот d.DNAME = 'Research' е услов за селекција, а додека пак условот d.DNUMBER = t.DNO е услов за спојување на торките.

Егзистенцијален и Универзален Квантификатор

- Најди ги имињата на вработените кои работат на сите проекти контролирани од одделот со број 5.

$$\{e.LNAME, e.FNAME \mid EMPLOYEE(e) \text{ AND } ((\forall x)(\text{not}(\text{PROJECT}(x)) \text{ OR } \text{not}(x.DNUM=5)) \text{ OR } ((\exists w)(\text{WORKS_ON}(w) \text{ AND } w.ESSN=e.SSN \text{ AND } x.PNUMBER=w.PNO))))\}$$

- Извади ги сите торки од универзалната квантификација кои не се од интерес преку користење на условот точно *за сите вакви торки*.
 - Првите торки кои се вадат (преку правење да се евалуираат автоматски на точно) се тие кои не се во релацијата R која ни е од интерес.
- Во горниот израз, преку изразот **not(PROJECT(x))** внатре во универзално квантифицираната формула евалуира во точно за сите торки x кои не се наоѓаат во PROJECT релацијата.
 - После ги вадиме торките кои не ни се од интерес од самата релација R. Изразот not(x.DNUM=5) евалуира во точно за сите торки x кои се наоѓаат во релцијата проект, но не се контролирани од одделот со број 5.
- Конечно, мора да специфицираме услов кој мора да ги држи заедно сите преостанати торки од релацијата R.

$$((\exists w)(\text{WORKS_ON}(w) \text{ AND } w.ESSN=e.SSN \text{ AND } x.PNUMBER=w.PNO))$$

Релациона пресметка врз домен

- Друга варијанта на релационата пресметка наречена релациона пресметка врз домен, или домен пресметка е еквивалентна со торка пресметката и со релационата алгебра.
- Домен пресметката се разликува од торка пресметката во типот на променливите кои се користат во формулите:
 - Наместо да се имаат променливи кои се простираат низ торките, овде имаме променливи кои се простираат врз вредностите од домените кои може да ги добијат атрибутите.
- За да се формира релација со степен n како резултат од прашалникот, мора да имаме n од овие доменски променливи — по една за секој атрибут

Релациона пресметка врз домен

➤ Изразот од домен пресметката е од следната форма

$$\{ x_1, x_2, \dots, x_n \mid \text{COND}(x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m}) \}$$

- каде $x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m}$ се доменските променливи кои се простираат врз домените (на атрибутите)
- и COND е условот или формулата на релационата пресметка врз доменот

Релациона пресметка врз домен

Најди ги датумот на раѓање и адресата на вработениот чие име е 'John B. Smith'.

$$\{uv \mid (\exists q) (\exists r) (\exists s) (\exists t) (\exists w) (\exists x) (\exists y) (\exists z) (\text{EMPLOYEE}(qrstuvwxyz) \text{ and } q='John' \text{ and } r='B' \text{ and } s='Smith')\}$$

- Кратенката **EMPLOYEE(qrstuvwxyz)** ги користи променливите без да ги одделува со записка:
EMPLOYEE(q,r,s,t,u,v,w,x,y,z)
- Десетте променливи од релацијата вработен се потребни, секоја посебно да го дефинира опсегот на вредности за секој од атрибутите.
 - Од десетте променливи q, r, s, \dots, z , само u и v се слободни.
- Се специфицираат потребните атрибути, BDATE и ADDRESS, преку слободните доменски променливи u за BDATE и v за ADDRESS.
- Специфицирај го условот за селекција на торка следен од вертикалната црта (|)
 - имено, секвенцата од вредностите придружени до променливите $qrstuvwxyz$ ќе бидат торка од вработен релацијата и вредностите за q (FNAME), r (MINIT), и s (LNAME) треба да бидат 'John', 'B', и 'Smith', соодветно.

Преглед на лекцијата

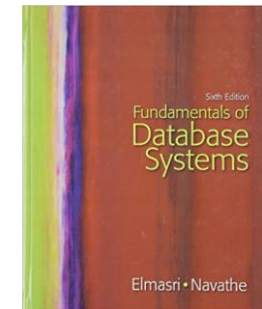
- Релациона алгебра
 - Единечни релациони операции
 - Операции од релационата алгебра од теоријата на множества
 - Бинарни релациони операции
 - Агрегатни функции и групирање на податоците
 - Дополнителни релациони операции
- Релациона пресметка
 - Релациона пресметка врз торка
 - Релациона пресметка врз домен

Користена литература



➤ **Глава 6** (173 - 221)

➤ **Глава 6** (145 - 194)



➤ **Глава 2** (51 - 100)

➤ **Глава 5** (247 - 284)