

Álgebra Relacional

Base de Dados - 2018/19
Carlos Costa

Introdução

Linguagem de Consulta/Interrogação de BD

- Álgebra Relacional
 - Linguagem formal do Modelo Relacional
 - Um conjunto básico de operações
- Outras linguagem formais: *relational calculus*
- As linguagens formais oferecem uma base teórica para a linguagem de consulta utilizada na prática.
- Linguagem prática do Modelo Relacional
 - SQL

Álgebra Relacional

Questões?

- Como deve ser uma linguagem de interrogação da BD?
- Como formular interrogações?
- Que tipo de interrogações existem?
- Como é que são os resultados?
- Expressões de álgebra relacional (linguagem).
 - Sequência de operações de álgebra relacional.
 - Permitem formular pedidos básicos de recuperação de informação (*retrieval requests*) sobre uma ou mais relações.
- Para formalizar uma interrogação necessitamos de um conjunto de operadores que operam sobre as relações e devolvem uma nova relação.
- Vamos estudar um conjunto de operações.

3

Álgebra Relacional - Operações Básicas

- Seleção



- Projeção



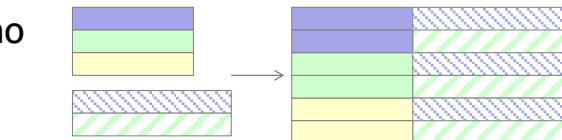
- União



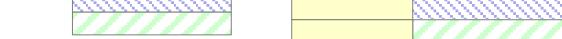
- Diferença



- Produto Cartesiano



- Renomeação



Seleção



- Notação: $\sigma_{<\text{selection condition}>}(R)$
 - Utilizada para selecionar um subconjunto de tuplos da relação ($t \in R$) que satisfazem os critérios de seleção.
 - “selection condition” é uma expressão booleana.

$\text{Relation2} \leftarrow \sigma_{<\text{selection condition}>}(\text{Relation1})$

- O resultado é uma nova relação (Relation2) que tem um esquema relacional igual à original (Relation1).

5

Seleção - Predicado

- Operadores de Comparação
 - Permitem comparar dois atributos ou um atributo com um valor.
 - Operandos: Nomes dos atributos e constantes.
 - Operadores: $=, \neq, \leq, \geq, <, >$
 - Exemplos:
 - $\sigma_{Dno=4}(\text{EMPLOYEE})$
 - $\sigma_{Salary>30000}(\text{EMPLOYEE})$

- Condições Booleanas
 - Utilização de AND, OR e NOT.
 - Exemplo:

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

6

Seleção - Exemplo

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

SQL query (próxima aula...)

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

**SELECT * FROM EMPLOYEE
WHERE Dno=4 AND Salary>25000
OR Dno=5 AND Salary>30000;**

Fname	Minit	Lname	Ssn	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

7

Projeção

- Notação: $\Pi_{<\text{attribute list}>}(\text{R})$
 - $<\text{attribute list}> = A_1, A_2, \dots, A_k$
 - $A_1 \dots A_k$ são nomes dos atributos da relação R
- O resultado é uma nova relação só com os k atributos selecionados.
- São removidas as linhas duplicadas do resultado.
 - Condição de conjunto (set)

8

deti

Projeção - Exemplo

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

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

SQL query:

```
SELECT DISTINCT Lname, Fname, Salary
FROM EMPLOYEE;
```

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

9

deti

Encadeamento de Operações

- $\Pi_{\text{Fname}, \text{Lname}, \text{Salary}}(\sigma_{\text{Dno}=5}(\text{EMPLOYEE}))$
- Se quisermos renomear os atributos e a relação:
 $\text{TEMP} \leftarrow \sigma_{\text{Dno}=5}(\text{EMPLOYEE})$
 $\text{R}(\text{First_name}, \text{Last_name}, \text{Salary}) \leftarrow \Pi_{\text{Fname}, \text{Lname}, \text{Salary}}(\text{TEMP})$

TEMP

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

R

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

10

Renomeação

- Notação: $\rho_{R_2(B_1, B_2, \dots, B_n)}(R_1)$ ou $\rho_{R_2}(R_1)$
- ou $\rho_{(B_1, B_2, \dots, B_n)}(R_1)$
- No primeiro caso o resultado é uma nova relação R2 com os atributos renomeados (B1, B2, ..., Bn).
- No segundo caso só renomeamos a relação.
- No terceiro só renomeamos os atributos.

SQL query:

```
SELECT E.Fname AS First_name, E.Lname AS Last_name, E.Salary AS Salary
FROM EMPLOYEE AS E
WHERE E.Dno=5;
```

Seleção

R1: EMPLOYEE
R2: E
Fname -> First_name
Lname -> Last_Name
...

11

União

- Notação: $R \cup S = \{t : t \in R \vee t \in S\}$
- As tabelas têm de ser compatíveis
 - Mesmo número de atributos
 - Atributos com domínios compatíveis
- O resultado é uma relação que inclui todos os tuplos de R e de S
 - Os tuplos duplicados são eliminados

STUDENT	Fn	Ln
Susan	Yao	
Ramesh	Shah	
Johnny	Kohler	
Barbara	Jones	
Amy	Ford	
Jimmy	Wang	
Ernest	Gilbert	

U

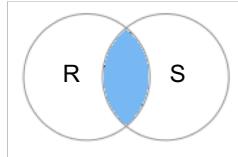
INSTRUCTOR	Fname	Lname
John	Smith	
Ricardo	Browne	
Susan	Yao	
Francis	Johnson	
Ramesh	Shah	

→

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

12

Intersecção



- Notação: $R \cap S = \{t : t \in R \wedge t \in S\}$
- As tabelas têm de ser compatíveis
 - Mesmo número de atributos
 - Atributos com domínios compatíveis
- O resultado é uma relação que inclui os tuplos que existem simultaneamente em R e S
 - Os tuplos duplicados são eliminados

STUDENT	
Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

\cap

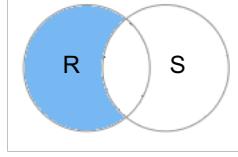
INSTRUCTOR	
Fname	Lname
John	Smith
Ricardo	Browne
Susan	Yao
Francis	Johnson
Ramesh	Shah

→

Fn	Ln
Susan	Yao
Ramesh	Shah

13

Diferença



- Notação: $R - S = \{t : t \in R \wedge t \notin S\}$
- As tabelas têm de ser compatíveis
 - Mesmo número de atributos
 - Atributos com domínios compatíveis
- O resultado é uma relação que inclui os tuplos de R que não existem em S

STUDENT	
Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

-

INSTRUCTOR	
Fname	Lname
John	Smith
Ricardo	Browne
Susan	Yao
Francis	Johnson
Ramesh	Shah

→

Fn	Ln
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

14

União, Intersecção e Diferença

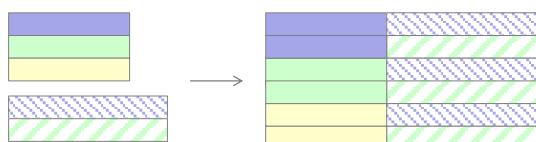
- Em SQL existem os seguintes comandos
 - UNION (ALL), INTERSECT (ALL) e EXCEPT (ALL)

Propriedades:

- União e Intersecção são operações comutativas:
 - $R \cup S = S \cup R$ e $R \cap S = S \cap R$
- A diferença não é comutativa:
 - $R - S \neq S - R$
- União e Intersecção são operações associativas:
 - $R \cup (S \cup T) = (R \cup S) \cup T$ e $(R \cap S) \cap T = R \cap (S \cap T)$

15

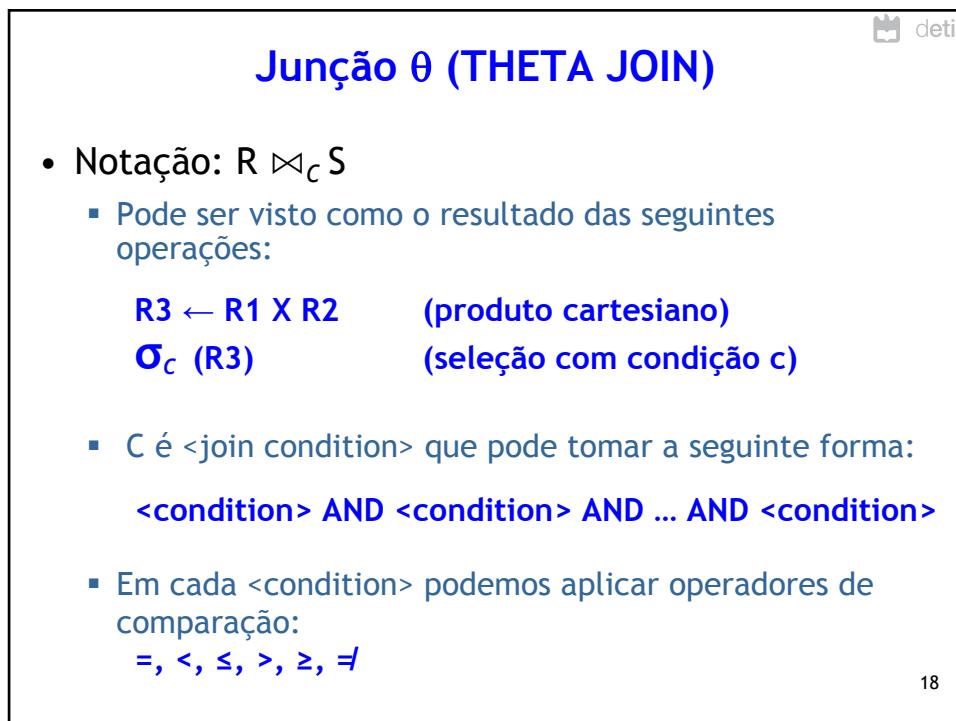
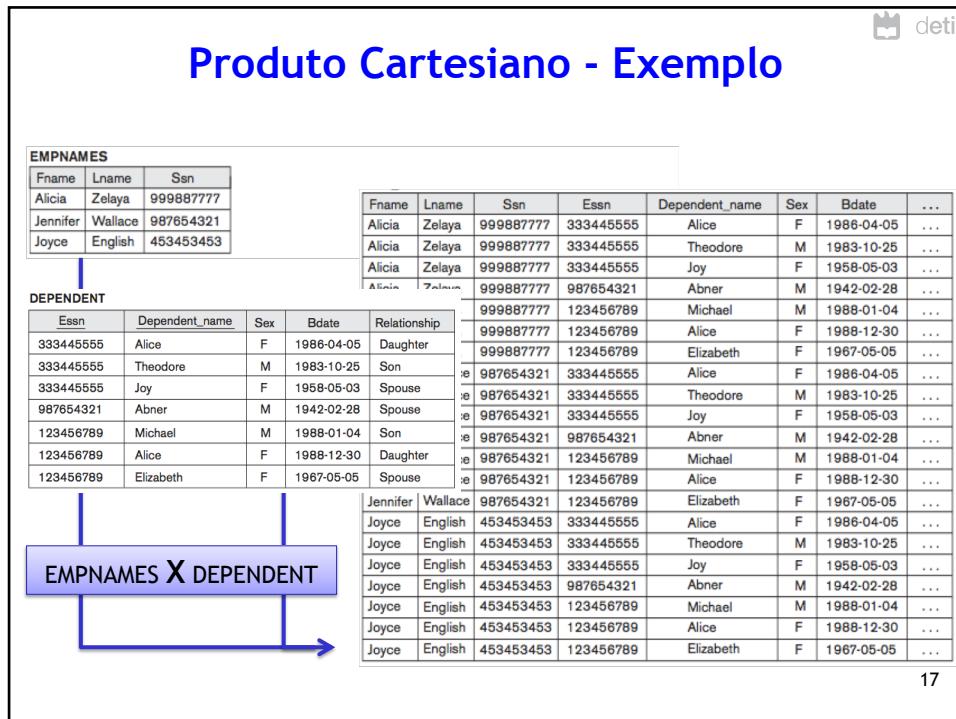
Produto Cartesiano



- Notação: $R \times S$
- Permite-nos combinar tuplos de relações diferentes.
 - O resultado é uma nova relação (Q) que combina cada elemento (tuplo) de uma relação (R) com um elemento (tuplo) da outra relação (S):

$$Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m) = R(A_1, A_2, \dots, A_n) \times S(B_1, B_2, \dots, B_m)$$
 - O número de tuplos de Q é $n * m$.
- UK: “CROSS JOIN”

16



Junção θ (THETA JOIN) - Exemplo

- Pretendemos saber os nomes dos funcionários gestores de departamentos

EMPLOYEE										
Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno	
John	B	Smith	123456789	1985-01-09	731 Fondren, Houston, TX	M	30000	333445555	5	
Franklin	T	Wong	333445555	1985-12-08	638 Voss, Houston, TX	M	40000	888665555	5	
Alicia	J	Zelya	999887777	1988-01-19	3321 Castle, Spring, TX	F	25000	987654321	4	
Jennifer	S	Wallace	987654321	1981-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4	
Ramesh	K	Narayan	666884444	1982-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	
Joyce	A	English	453453453	1972-07-31	5831 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	1987-11-10	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

Para obter o nome dos gestores temos de combinar cada tuplo do departamento (Department) com um tuplo dos funcionários (Employee) cujo Ssn é igual ao Mgr_ssn.

$\text{DEPT_MGR} \leftarrow \text{DEPARTMENT} \bowtie_{\text{Mgr_ssn}=\text{Ssn}} \text{EMPLOYEE}$

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	...

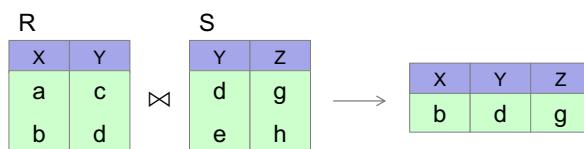
Depois só temos de utilizar projeção para obter os atributos desejados:

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

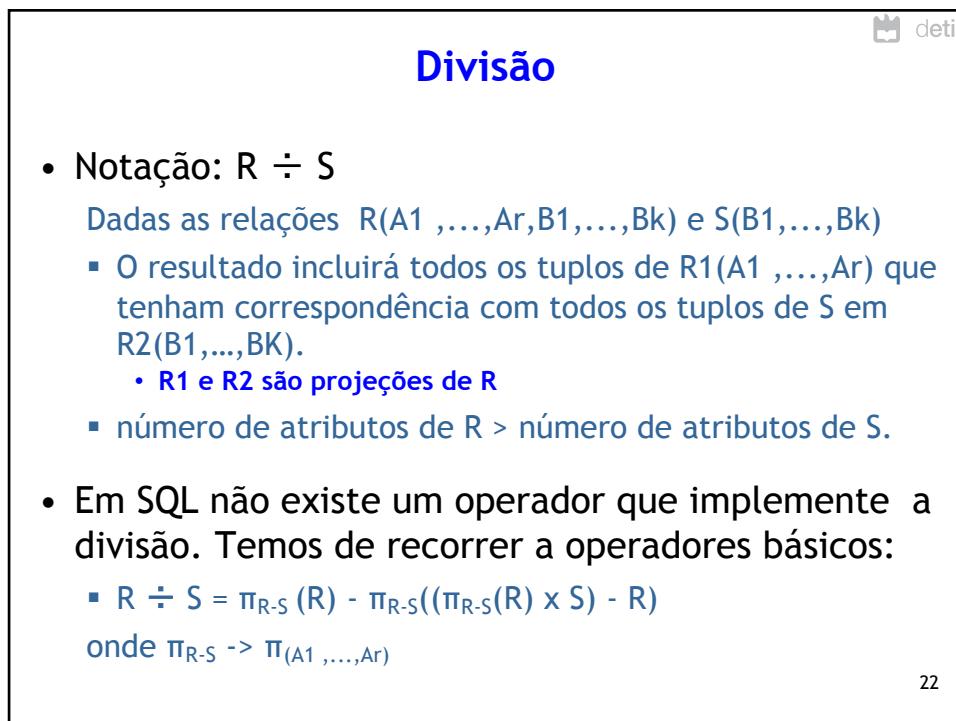
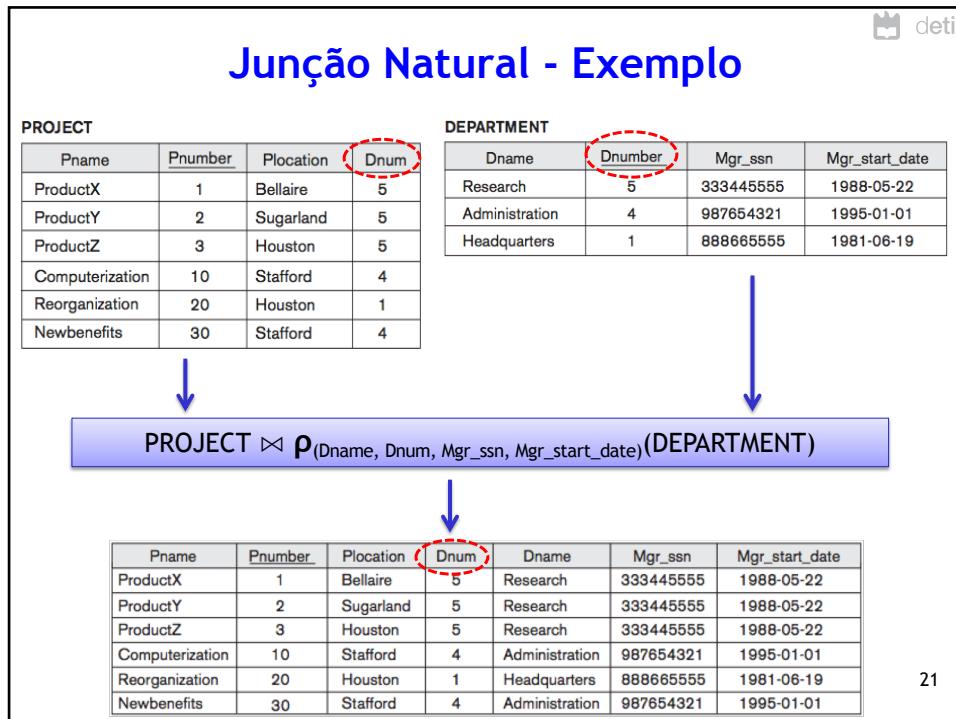
19

Junção - Variações da Junção θ

- Equi-Junção (EquiJoin)**
 - É utilizado o operador $=$ na condição de junção.
 - Exemplo anterior: $\text{DEPARTMENT} \bowtie_{\text{Mgr_ssn}=\text{Ssn}} \text{EMPLOYEE}$.
 - Vamos ter sempre duas colunas repetidas.
- Junção Natural (Natural Join): $R \bowtie S$**
 - Condição implícita: igualdade dos atributos com o mesmo nome.
 - Os atributos repetidos são removidos.
 - Nota: Muitas vezes opta-se por renomear colunas de modo a facilitar junções naturais.



20



Divisão - Exemplos

$$R \div S = T$$

Department

Dno	Name	Location
1	Research	Houston
2	Commercial	Bellaire
3	Administration	LA
2	Commercial	Houston
4	Headquarters	Bellaire
2	Commercial	LA

÷

Location

Location
Houston
Bellaire
LA

→

Dno	Name
2	Commercial

Departamentos que existem em todas as localizações?

23

Operações Álgebra Relacional - Resumo

OPERATION	PURPOSE	NOTATION
SELECT	Selects all tuples that satisfy the selection condition from a relation R .	$\sigma_{<\text{selection condition}>}(R)$
PROJECT	Produces a new relation with only some of the attributes of R , and removes duplicate tuples.	$\pi_{<\text{attribute list}>}(R)$
THETA JOIN	Produces all combinations of tuples from R_1 and R_2 that satisfy the join condition.	$R_1 \bowtie_{<\text{join condition}>} R_2$
EQUIJOIN	Produces all the combinations of tuples from R_1 and R_2 that satisfy a join condition with only equality comparisons.	$R_1 \bowtie_{<\text{join condition}>} R_2$, OR $R_1 \bowtie_{<\text{join attributes 1>},<\text{join attributes 2>}} R_2$
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.	$R_1 *_{<\text{join conditions}>} R_2$, OR $R_1 *_{<\text{join attributes 1>},<\text{join attributes 2>}} R_2$, OR $R_1 *_{R_2} R_2$
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 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 $Z = X \cup Y$.	$R_1(Z) \div R_2(Y)$

24

Álgebra Relacional - Operações Estendidas

- Semi-Join (Semi Junção)
 - Left Semi Join
 - Right Semi Join
- Outer Join (Junção Externa)
 - Left Outer Join
 - Right Outer Join
 - Full Outer Join
- Agregação
 - Funções de Agregação

25

Semi Join

- Left Semi Join: $R \bowtie S = \Pi_R(R \bowtie S)$

Projeção dos atributos de R na junção natural de R com S

R	S	
x	y	
a	c	
b	d	
		→
x	y	
b	d	

- Right Semi Join: $R \ltimes S = \Pi_S(R \bowtie S)$

Projeção dos atributos de S na junção natural de R com S

R	S	
x	y	
a	c	
b	d	
		→
y	z	
d	g	

26

Inner Join vs Outer Join

Inner Join

- As operações de junção anteriores combinam dados de duas tabelas para que estes possam ser apresentados na forma de uma única tabela.
- Os tuplos que não estão relacionados (*matching*) são descartados.
 - Incluindo os tuplos com valores Null nos atributos de junção.

Outer Join

- Incluímos no resultado todos os tuplos de uma (ou de ambas) das relações componentes.
- Os atributos que não fazem *matching* são preenchidos com *Null*.

27

Outer Join

- Left Outer Join: $R \bowtie S$

R		S			Result			
A1	A2	B1	B2	$\bowtie_{A2=B1}$	A1	A2	B1	B2
a	c	d	g		a	c	null	null
b	d	e	h		b	d	d	g

- Right Outer Join: $R \bowtie S$

R		S			Result			
A1	A2	B1	B2	$\bowtie_{A2=B1}$	A1	A2	B1	B2
a	c	d	g		b	d	d	g
b	d	e	h		null	null	e	h

- Full Outer Join: $R \bowtie S$

R		S			Result			
A1	A2	B1	B2	$\bowtie_{A2=B1}$	A1	A2	B1	B2
a	c	d	g		a	c	null	null
b	d	e	h		b	d	d	g

28

Left Outer Join - Exemplo

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	1956-12-08	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	886884444	1982-09-15	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

$\Pi_{Fname, Minit, Lname, Dname} (EMPLOYEE \bowtie_{Ssn=Mgr_ssn} DEPARTMENT)$

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

29

Join - Quadro Resumo

- Natural | Left Outer | Right Outer | Full Outer

X	Y
1	
2	
3	
4	
5	

Y	Z
	a
	b
	c
	d

X	Y	Z
1		null
2		a
3		null
4		b
4		c
5		null
null		d

X	Y	Z
x	v	x
v	v	v
x	v	x
v	v	v
v	v	v
x	v	x
x	x	v

30

Agregação

- Operação de Agregação

<grouping attributes>  <function list> (R)

 - Script F
symbol

- Operações sobre vários tuplos da relação

- Lista de Funções de Agregação:

- avg: média dos valores
- min: mínimo dos valores
- max: máximo dos valores
- sum: soma dos valores
- count: número dos valores

31

Funções de Agregação

- Também podem ser usadas em projeções
 - criar atributos agregados
 - os atributos não agregados são agrupados de forma a não haver valores repetidos.
- Exemplos:

$\Pi_{A1, A2, M = \text{avg}(A3)} (R)$

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

$\Pi_{Dno, Avg_Salary=\text{avg}(Salary)}(\text{EMPLOYEE})$

Dno	Avg_Salary
1	55000
4	31000
5	33250

32

Agregação (Grouping) - Exemplos

EMPLOYEE

Name	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

$\exists \text{ count}(\text{Ssn}), \text{ avg}(\text{Salary})(\text{EMPLOYEE})$

Count_ssn	Average_salary
8	35125

$Dno \exists \text{ count}(\text{Ssn}), \text{ avg}(\text{Salary})(\text{EMPLOYEE})$

Dno	Count_ssn	Average_salary
5	4	33250
4	3	31000
1	1	55000

$\rho_{R(Dno, \text{No_of_employees}, \text{Average_sal})}(Dno \exists \text{ count}(\text{Ssn}), \text{ avg}(\text{Salary})(\text{EMPLOYEE}))$

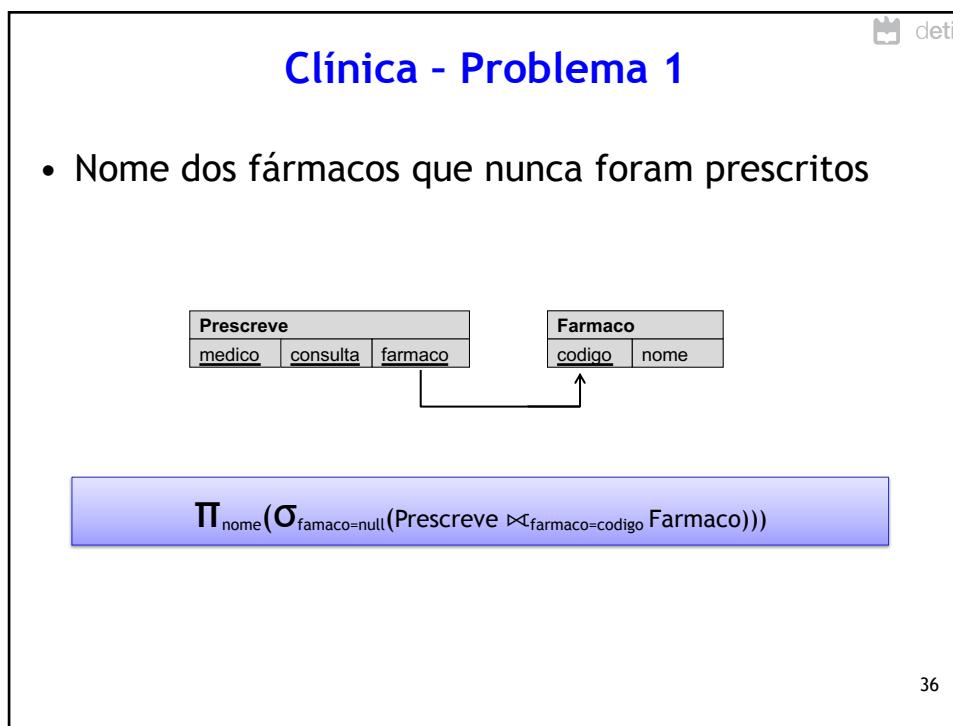
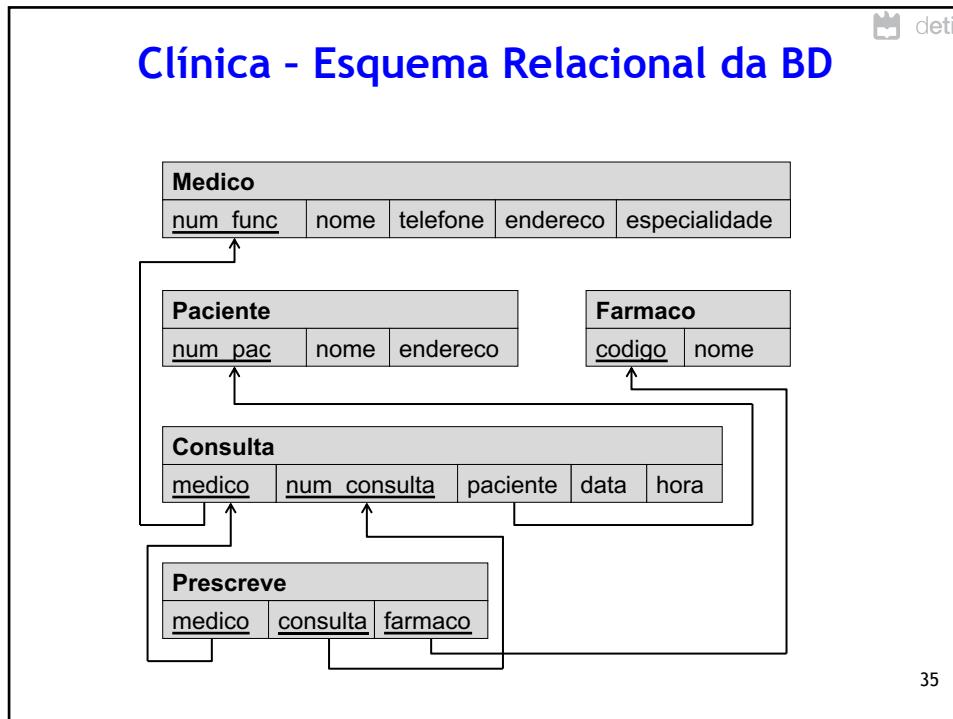
Dno	No_of_employees	Average_sal
5	4	33250
4	3	31000
1	1	55000

33

Álgebra Relacional - Queries Caso de Estudo

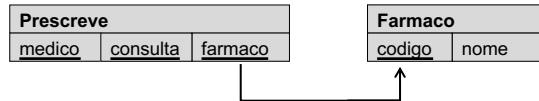
Clínica Médica

34



Clínica - Problema 2

- O número de fármacos prescritos em cada consulta



$\Pi_{medico, consulta, num_farm=count(farmaco)} (\text{Prescreve})$

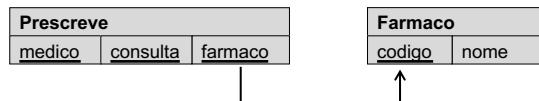
Ou

$\text{medico, consulta } \Sigma_{count(farmaco)} (\text{Prescreve})$

37

Clínica - Problema 3

- Para cada médico, a quantidade média de fármacos receitados por consulta



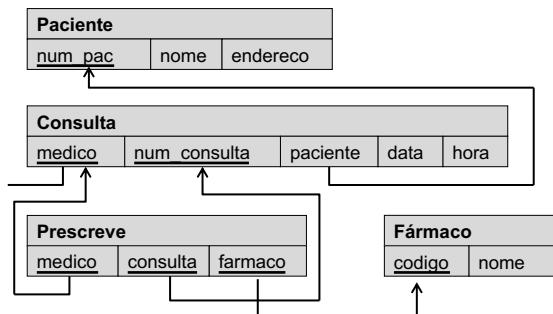
$\text{temp} \leftarrow \Pi_{medico, consulta, num_farm=count(farmaco)} (\text{Prescreve})$

$\Pi_{medico, avg_farmaco=avg(num_farm)} (\text{temp})$

38

Clínica - Problema 4

- O nome de todos os fármacos prescritos, incluindo a quantidade, para o paciente número 35312161



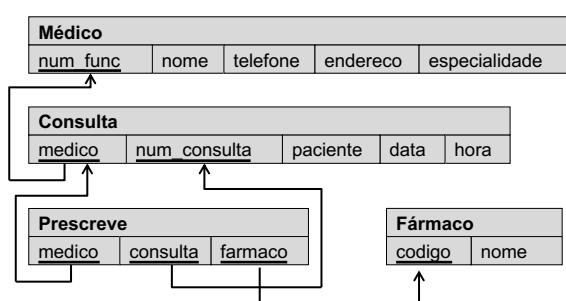
$\text{temp} \leftarrow \Pi_{\text{medico}, \text{num_consulta}} (\sigma_{\text{paciente}=35312161} (\text{Consulta}))$

$\text{temp2} \leftarrow \Pi_{\text{farmaco}, \text{quantidade}=\text{count}(\text{farmaco})} (\text{temp} \bowtie_{\text{medico}=\text{medico} \text{ AND } \text{num_consulta}=\text{consulta}} \text{Prescreve})$

$\Pi_{\text{nome}, \text{quantidade}} (\text{temp2} \bowtie_{\text{farmaco}=\text{codigo}} \text{Farmaco})$

Clínica - Problema 5

- O nome dos fármacos que já foram prescritos por todos os médicos da clínica



$\text{temp} \leftarrow (\Pi_{\text{farmaco}, \text{medico}} (\text{Prescreve})) \div (\rho_{\text{medico}} (\Pi_{\text{num_func}} (\text{Médico})))$

$\Pi_{\text{nome}} (\rho_{\text{codigo}, \text{medico}} (\text{temp}) \bowtie \text{Farmaco})$

A Seguir?

Data Operations – Relational Algebra		SQL – Data Manipulation															
<table border="1"> <thead> <tr> <th>r</th> <th>s</th> <th>r ∪ s</th> </tr> </thead> <tbody> <tr> <td>A B</td> <td>A B</td> <td>A B</td> </tr> <tr> <td>α 1</td> <td>α 2</td> <td>α 1</td> </tr> <tr> <td>α 2</td> <td>β 3</td> <td>β 1</td> </tr> <tr> <td>β 1</td> <td></td> <td>β 3</td> </tr> </tbody> </table> <p>Query syntax: SELECT <desired attributes> FROM <one or more tables> WHERE <predicate holds for selected tuple> GROUP BY <key columns, aggregations> HAVING <predicate holds for selected group> ORDER BY <columns to sort></p>	r	s	r ∪ s	A B	A B	A B	α 1	α 2	α 1	α 2	β 3	β 1	β 1		β 3		<p>SQL query:</p> <pre>SELECT Pnumber, Pname, COUNT (*) FROM PROJECT, WORKS_ON WHERE Pnumber=Pno GROUP BY Pnumber, Pname;</pre> <p>SQL query:</p> <pre>INSERT INTO EMPLOYEE (Fname, Lname, Ssn, Dno) VALUES('Robert', 'Hatcher', '980760540', 2);</pre>
r	s	r ∪ s															
A B	A B	A B															
α 1	α 2	α 1															
α 2	β 3	β 1															
β 1		β 3															
<p>SQL – Describe Database Schema</p> <pre>CREATE TABLE DEPARTMENT (Dname VARCHAR(15) NOT NULL, Dnumber INT NOT NULL, Mgr_ssn CHAR(9) NOT NULL, Mgr_start_date DATE, PRIMARY KEY (Dnumber), UNIQUE (Dname), FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn);</pre>	<p>The Relational Schema</p> <p>Part (Name, Description, Part#) Supplier (Name, Addr) Customer (Name, Addr) Supplies (Name, Part#, Date) Orders (Name, Part#)</p>																

Resumo

Álgebra Relacional:

- Operações Básicas
- Operações Estendidas
- Caso de Estudo - Queries