

*Corso di Laurea in Ingegneria Gestionale
SAPIENZA Università di Roma
Esercitazioni del corso di Basi di Dati
Prof.ssa Catarci e Prof.ssa Scannapieco*

Anno Accademico 2011/2012

2 – Algebra Relazionale

Francesco Leotta

Ultimo aggiornamento : 12/03/2012

L'algebra relazionale

- ▶ **Algebra Relazionale**: linguaggio *procedurale* (specifica come viene generato il risultato) di *interrogazione*
- ▶ Costituita da un insieme di operatori:
 - ▶ definiti su relazioni
 - ▶ che producono relazioni
 - ▶ e possono essere composti per formulare interrogazioni complesse
- ▶ **Operatori** dell'algebra relazionale:
 - ▶ Insiemistici (unione, intersezione, differenza)
 - ▶ Ridenominazione
 - ▶ Prodotto cartesiano
 - ▶ Selezione
 - ▶ Proiezione
 - ▶ Join (naturale, equi-join, condizionale o theta-join)
 - ▶ Divisione

Operatori insiemistici

- ▶ Una relazione è un *insieme di tuple omogenee* (cioè definite sugli *stessi attributi*)
- ▶ E' possibile applicare gli operatori insiemistici **solo** a relazioni per cui valga la proprietà di ***compatibilità rispetto all'unione***:
 - ▶ *le relazioni in ingresso hanno lo stesso numero di campi*
 - ▶ *campi corrispondenti delle due relazioni, presi in ordine da sinistra a destra, hanno lo stesso dominio (la compatibilità non si verifica con i nomi degli attributi)*
- ▶ Se due relazioni R1 e R2 sono compatibili rispetto all'unione, ma hanno i nomi degli attributi differenti, la relazione che si ottiene come risultato eredita per convenzione i nomi di R1



ATTENZIONE :

Si rischia di ottenere un insieme di tuple disomogenee...

Operatori insiemistici - Unione

Laureati

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38

Dirigenti

Matricola	Cognome	Età
9297	Neri	56
7432	Neri	39
9824	Verdi	38

U

Laureati U Dirigenti

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38
9297	Neri	56



- L'unione di due relazioni r_1 e r_2 è indicata con $r_1 \cup r_2$ e contiene tutte le tuple (**prese una sola volta**) presenti in r_1 , in r_2 oppure in entrambe
- Lo schema del risultato è identico allo schema di r_1

Operatori insiemistici - Intersezione

Laureati

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38



Dirigenti

Matricola	Cognome	Età
9297	Neri	56
7432	Neri	39
9824	Verdi	38

Laureati \cap Dirigenti

Matricola	Cognome	Età
7432	Neri	39
9824	Verdi	38



- L'intersezione di due relazioni r_1 e r_2 è indicata con $r_1 \cap r_2$ e contiene tutte le tuple (prese una sola volta) presenti contemporaneamente sia in r_1 che in r_2
- Lo schema del risultato è identico allo schema di r_1

Operatori insiemistici - Differenza

Laureati

Matricola	Cognome	Età
7274	Rossi	37
7432	Neri	39
9824	Verdi	38

Dirigenti

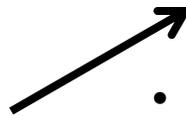
Matricola	Cognome	Età
9297	Neri	56
7432	Neri	39
9824	Verdi	38

—

Laureati - Dirigenti

Matricola	Cognome	Età
7274	Rossi	37

- La differenza tra due relazioni r_1 e r_2 è indicata con $r_1 - r_2$ e contiene tutte le tuple presenti in r_1 ma non in r_2
- Lo schema del risultato è identico allo schema di r_1



Un'unione sensata...ma impossibile

Paternità

Padre	Figlio
Francesco	Maurizio
Francesco	Carla
Giuseppe	Maria

U

Maternità

Madre	Figlio
Adriana	Maurizio
Adriana	Paolo
Eleonora	Maria

Paternità U Maternità ??

L'unione rispetta la proprietà di compatibilità rispetto all'unione, ma il risultato è disomogeneo...

Padre	Figlio
Francesco	Maurizio
Francesco	Carla
Giuseppe	Maria
Adriana	Maurizio
Adriana	Paolo
Eleonora	Maria

La ridenominazione 1\3

- ▶ Sintassi dell'operatore : $\rho(R(F),E)$
 - ▶ **E** : espressione arbitraria di algebra relazionale (*es. Paternità U Maternità*)
 - ▶ **R** : istanza di relazione che rappresenta il risultato della esecuzione di E (contiene le stesse tuple di E, ma con alcuni campi rinominati)
 - ▶ **F** : lista di “ridenominazione” nella forma:
 - ▶ *Vecchio nome* \rightarrow *Nuovo nome*
 - ▶ *Posizione* \rightarrow *Nuovo nome*
- ▶ $\rho(C(\text{Padre} \rightarrow \text{Genitore}), \text{Paternità U Maternità})$
oppure
- ▶ $\rho(C(1 \rightarrow \text{Genitore}), \text{Paternità U Maternità})$

Genitore	Figlio
Francesco	Maurizio
Francesco	Carla
Giuseppe	Maria
Adriana	Maurizio
Adriana	Paolo
Eleonora	Maria

$C(\text{Genitore:String}, \text{Figlio:String}) \longrightarrow$

La ridenominazione 2\3

Per ottenere la relazione Genitore-Figlio, si può anche scrivere alternativamente....

Paternità

Padre	Figlio
Francesco	Maurizio
Francesco	Carla
Giuseppe	Maria

$\rho((Padre \rightarrow Genitore), Paternità)$



Genitore	Figlio
Francesco	Maurizio
Francesco	Carla
Giuseppe	Maria

Maternità

Madre	Figlio
Adriana	Maurizio
Adriana	Paolo
Eleonora	Maria

$\rho((Madre \rightarrow Genitore), Maternità)$



Genitore	Figlio
Adriana	Maurizio
Adriana	Paolo
Eleonora	Maria

$\rightarrow \rho((Padre \rightarrow Genitore), Paternità) \cup \rho((Madre \rightarrow Genitore), Maternità) \leftarrow$

La ridenominazione 3\3

Impiegati

Cognome	Ufficio	Stipendio
Rossi	Roma	55
Neri	Milano	64

Operai

Cognome	Fabbrica	Salario
Bruni	Monza	45
Verdi	Latina	55

$\rho((\text{Ufficio} \rightarrow \text{Sede}, \text{Stipendio} \rightarrow \text{Retribuzione}), \text{Impiegati})$

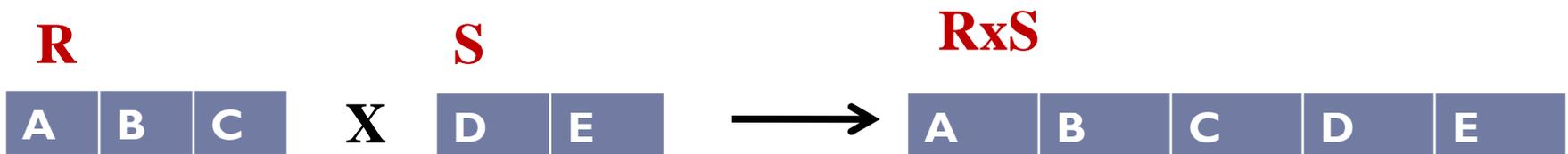
U

$\rho((\text{Fabbrica} \rightarrow \text{Sede}, \text{Salario} \rightarrow \text{Retribuzione}), \text{Operai})$

Cognome	Sede	Retribuzione
Rossi	Roma	55
Neri	Milano	64
Bruni	Monza	45
Verdi	Latina	55

Il prodotto cartesiano 1\3

- ▶ E' anch'esso un operatore insiemistico
- ▶ Per essere utilizzato **non richiede che sia valida** la proprietà di *compatibilità rispetto all'unione*
- ▶ Sintassi dell'operatore : **$R \times S$**
- ▶ restituisce un'istanza di relazione il cui schema contiene tutti i campi di R (nell'ordine originale) seguiti da tutti i campi di S (nell'ordine originale)



Il prodotto cartesiano 2\3

- ▶ contiene una tupla $\langle r,s \rangle$ per ogni coppia di tuple $r \in R$ e $s \in S$
- ▶ contiene un numero di tuple pari al prodotto delle cardinalità degli operandi

Impiegati

Impiegato	Codice
Rossi	A
Neri	B
Bianchi	B

Reparti

Capo	Codice
Mori	A
Bruni	B

ATTENZIONE : Se due relazioni hanno degli attributi con nomi in comune, i campi corrispondenti nel prod.cartesiano non hanno nome (per convenzione) e ci si può riferire a loro solo attraverso la posizione che occupano nello schema...

Impiegati X Reparti

Impiegato	(Codice)	Capo	(Codice)
Rossi	A	Mori	A
Rossi	A	Bruni	B
Neri	B	Mori	A
Neri	B	Bruni	B
Bianchi	B	Mori	A
Bianchi	B	Bruni	B

→ conflitto di nomi



si può risolvere con una ridenominazione...

Il prodotto cartesiano 3\3

$\rho(\text{nuovaRelazione}(2 \rightarrow \text{CodImpiegato}, 4 \rightarrow \text{CodReparto}), \text{Impiegati} \times \text{Reparti})$

nuovaRelazione



In questo caso è necessario sfruttare la notazione posizionale...

Impiegato	CodImpiegato	Capo	CodReparto
Rossi	A	Mori	A
Rossi	A	Bruni	B
Neri	B	Mori	A
Neri	B	Bruni	B
Bianchi	B	Mori	A
Bianchi	B	Bruni	B

conflitto di nomi risolto

Selezione 1\2

- ▶ Operatore unario per **selezionare righe da una relazione**
- ▶ **Sintassi** : $\sigma_{\text{condizione}}(\mathbf{R})$
 - ▶ condizione di selezione = espressione booleana del tipo:
 - ▶ *attributo* **OP** *costante* oppure *attributo1* **OP** *attributo2*
 - ▶ **OP** = { < , <= , > , >= , = , ≠ }
 - ▶ Espandibile con i connettivi logici **V** , **^**
- ▶ **Semantica** :
 - ▶ la relazione risultato ha gli stessi attributi dell'operando e contiene le tuple dell'operando che soddisfano la condizione specificata

Esempio

Tutti gli Impiegati con codice = A

$\sigma_{\text{Codice}='A'}(\text{Impiegati})$

Impiegati

Impiegato	Codice
Rossi	A
Neri	B
Bianchi	B

Impiegato	Codice
Rossi	A

Selezione 2\2

▶ Esempio

- ▶ Tutti i Laureati con Cognome = 'Rossi' ed Età > 37

$\sigma_{\text{Cognome}='Rossi' \wedge \text{Età}>37}(\text{Laureati})$

▶ Esempio

- ▶ Tutti i Laureati con lo stesso Cognome dell'Università in cui hanno studiato

$\sigma_{\text{Cognome}=\text{Università}}(\text{Laureati})$

Laureati

Matricola	Cognome	Università	Età
7274	Rossi	Roma	37
7432	Rossi	Roma	39
9824	Roma	Roma	38

Matricola	Cognome	Università	Età
7432	Rossi	Roma	39

Matricola	Cognome	Università	Età
9824	Roma	Roma	38

Proiezione

▶ Operatore unario per **estrarre colonne da una relazione**

▶ Sintassi : $\Pi_{\text{lista_di_Attributi}}(\mathbf{R})$

▶ Semantica :

▶ la relazione risultato ha i soli attributi contenuti in *ListaAttributi* e contiene le tuple ristrette agli attributi nella lista (senza duplicati)

▶ **Esempio**

▶ **Cognome e Università
di tutti i laureati**

Laureati

Matricola	Cognome	Università	Età
7274	Rossi	Roma	37
7432	Rossi	Roma	39
9824	Roma	Roma	38

ATTENZIONE :
gli attributi non
contenuti nella lista
vengono proiettati fuori

$\Pi_{\text{Cognome, Università}}(\mathbf{Laureati})$



Cognome	Università
Rossi	Roma
Roma	Roma

Selezione e Proiezione

- ▶ Gli operatori di selezione e proiezione si possono combinare efficientemente

- ▶ **Esempio**

- ▶ **Cognome e Università dei laureati con più di 37 anni**

Laureati

Matricola	Cognome	Università	Età
7274	Rossi	Roma	37
7432	Rossi	Roma	39
9824	Roma	Roma	38

Cognome	Università
Rossi	Roma
Roma	Roma

$$\Pi_{\text{Cognome, Università}} (\sigma_{\text{Età} > 37}(\text{Laureati}))$$

Join condizionale

▶ **JOIN** = Operatore molto usato per **combinare informazioni tra due o più relazioni**

▶ **JOIN condizionale:**

▶ Sintassi e semantica:

▶ $R_1 \bowtie_c R_2$ equivale a $\sigma_c (R_1 \times R_2)$

▶ La *condizione di JOIN* è identica (nella forma) alla *condizione di selezione*

Un ***JOIN condizionale*** tra R_1 e R_2 è fondamentalmente una **SELECT** applicata sul prodotto cartesiano tra R_1 e R_2

R₁

Impiegato	Stipendio
Rossi	20
Neri	20
Bianchi	30

R₂

Capo	Stipendio
Mori	10
Bruni	20

R₁ $\bowtie_{R_1.Stipendio > 20}$ **R₂**

Impiegato	(Stipendio)	Capo	(Stipendio)
Bianchi	30	Mori	10
Bianchi	30	Bruni	20

Equi-Join

▶ **EQUI-JOIN:**

- ▶ Se la condizione di JOIN è composta solo da uguaglianze (eventualmente connesse da \wedge)
 - ▶ Ambo i lati di ogni condizione di uguaglianza debbono **riferirsi** ad un attributo

R₁

Impiegato	Stipendio
Rossi	20
Neri	20
Bianchi	30

R₂

Capo	Stipendio
Mori	10
Bruni	20

R₁  **R₂**
R1.Stipendio=R2.Stipendio

Impiegato	Stipendio	Capo
Rossi	20	Bruni
Neri	20	Bruni

- ▶ Lo schema di un risultato di un EQUI-JOIN contiene tutti gli **attributi** di R1 e quegli attributi di R2 che non compaiono nella condizione di JOIN
 - ▶ Solo un attributo Stipendio appare nel risultato (mantenerli entrambe è ridondante)

Join naturale 1\2

- ▶ E' un EQUI-JOIN in cui le uguaglianze sono specificate su tutti gli attributi aventi lo stesso nome in R1 e R2
- ▶ Le tuple del risultato sono ottenute combinando le tuple degli operandi con **valori uguali sugli attributi comuni**

Esempio : Un JOIN completo (ogni ennupla contribuisce al risultato)

R₁

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

R₂

Capo	Reparto
Mori	A
Bruni	B

R₁ ⋈ R₂

Impiegato	Reparto	Capo
Rossi	A	Mori
Neri	B	Bruni
Bianchi	B	bruni

- ▶ Se le due relazioni non hanno attributi in comune, il JOIN naturale si riduce ad essere un semplice prodotto cartesiano

Join naturale 2\2

▶ Esempio : Un JOIN non completo

R₁

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

R₂

Capo	Reparto
Mori	B
Bruni	C

R₁ ⋈ R₂

Impiegato	Reparto	Capo
Neri	B	Mori
Bianchi	B	Mori

▶ Esempio : Un JOIN vuoto

R₁

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

R₂

Capo	Reparto
Mori	D
Bruni	C

R₁ ⋈ R₂

Impiegato	Reparto	Capo
-----------	---------	------

Esercizio - Massimo e Minimo Assoluto

- ▶ Dato $R(\text{Impiegato}, \text{Stipendio})$, trovare il minimo stipendio in R .

Nella seconda parte dell'interrogazione vengono trovati tutti quei valori di **Stipendio** che non sono il minimo. Per far questo viene fatto un join tra la relazione R e se stessa, ridenominando tutti gli attributi della copia di R . La condizione di join seleziona quei valori dell'attributo **Stipendio** che sono maggiori di almeno un valore della sua controparte rinominata.

$\Pi_{\text{Stipendio}}(R) -$

$\Pi_{\text{Stipendio}}(R \bowtie$

$R.\text{Stipendio} > R1.\text{Stip}$

$(\rho(R1(1 \rightarrow \text{Imp}, 2 \rightarrow \text{Stip}), R))$

In questo modo vengono mantenute tutte le tuple tranne quella in cui l'attributo **Stipendio** assume il valore minore. Quindi per il "principio di complementarità" sottraendo dall'insieme iniziale l'insieme delle tuple dove Stipendio non è il minimo, otteniamo proprio il valore minimo che cercavamo.

→

Stipendio
10

R

Impiegato	Stipendio
Rossi	10
Neri	20
Bianchi	30

Esercizio - Massimo e Minimo Relativo

- ▶ Dato $R(\text{Studente}, \text{Esame}, \text{Voto})$, trovare il massimo voto per ogni studente in R .

E' molto simile al caso precedente. Il join in questo caso seleziona tutti i valori minimi di **Voto** per ogni valore dell'attributo **Studente**.

$\Pi_{\text{Studente}, \text{Voto}}(R) -$

$\Pi_{\text{Studente}, \text{Voto}}(R \bowtie$

$R.Voto < R1.Vot \text{ AND } R.Studente = R1.Stud$

$(\rho(R1(1 \rightarrow \text{Stud}, 2 \rightarrow \text{Exam}, 3 \rightarrow \text{Vot}), R))$))

R

Studente	Esame	Voto
Rossi	Analisi	18
Rossi	Basi di Dati	20
Bianchi	Analisi	30



Studente	Voto
Rossi	20
Bianchi	30

Esercizio - Cardinalità

- ▶ Dato $R(\text{Nome}, \text{Cognome}, \text{CF})$, trovare le persone omonime (stesso nome e cognome, ma diverso CF).

Viene ancora fatto un join tra la relazione R e se stessa, con però tutti gli attributi ridenominati. Il join permette di mantenere tutte quelle tuple in cui i valori degli attributi **Nome** e **Cognome** sono uguali alla loro controparte ridenominata, mentre il valore dell'attributo **CF** è diverso.

$$\Pi_{\text{Nome}, \text{Cognome}, \text{CF}}(R \bowtie_{\text{R.Nome}=\text{R1.Nom AND R.Cognome}=\text{R1.Cog AND R.CF} \neq \text{R1.Cod}} (\rho(\text{R1}(1 \rightarrow \text{Nom}, 2 \rightarrow \text{Cog}, 3 \rightarrow \text{Cod}), R)))$$

Nome	Cognome	CF
Marco	Rossi	AAAAA
Marco	Rossi	BBBBBB
Andrea	Bianchi	CCCCC



Nome	Cognome	CF
Marco	Rossi	AAAAA
Marco	Rossi	BBBBB

Divisione

- ▶ Non supportata come operatore primitivo, ma utile per alcuni tipi di interrogazione
- ▶ Sia A una relazione con due campi, x e y ; sia B una relazione con il solo campo y :
- ▶ $A/B = \{ \langle x \rangle \mid \text{per ogni } \langle y \rangle \in B, \langle x, y \rangle \in A \}$
 - ▶ cioè, A/B contiene tutte le tuple x tali che **per ogni** tupla y in B , ci sia una tupla xy in A
- ▶ E' possibile estendere il concetto a relazioni con un numero generico di campi

ESEMPIO : Trovare gli **Impiegati** *che lavorano in tutti i reparti*

R₁

Impiegato	Reparto
Rossi	A
Rossi	B
Bianchi	B

R₂

Reparto
A
B

R₁ / R₂

Impiegato
Rossi

Condizioni di selezione e valori NULL

Impiegati

Matricola	Cognome	Università	Età
7274	Rossi	Roma	37
7432	Rossi	Roma	41
9824	Roma	Roma	NULL

$\sigma_{Eta > 40}(\text{Impiegati})$???

La condizione è vera solo per valori **NON NULLI**

- ▶ Per riferirsi a valori NULLI esistono forme apposite di condizioni

IS NULL
IS NOT NULL

$\sigma_{Eta > 40} \vee \sigma_{Eta \text{ IS NULL}}(\text{Impiegati})$

Matricola	Cognome	Università	Età
7432	Rossi	Roma	41
9824	Roma	Roma	NULL

Esempi – Relazioni di riferimento

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Supervisione

Impiegato	Capo
7309	5698
5998	5698
9553	4076
5698	4076
4076	8123

Esempio 1

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Trovare *Matricola, Nome, Età, Stipendio* degli impiegati che guadagnano più di 40 milioni

$\sigma_{\text{Stipendio} > 40}(\text{Impiegati})$



Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Esempio 2

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Trovare *Matricola, Nome, Età* degli impiegati che *guadagnano più di 40 milioni*

Matricola	Nome	Età
7309	Rossi	34
5698	Bruni	43
4076	Mori	45
8123	Lupi	46

$\Pi_{\text{Matricola, Nome, Età}} (\sigma_{\text{Stipendio} > 40}(\text{Impiegati}))$



Esempio 3 – 1\4

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Supervisione

Impiegato	Capo
7309	5698
5998	5698
9553	4076
5698	4076
4076	8123

Trovare le *Matricole* dei capi degli impiegati che *guadagnano più di 40 milioni*



Una *buona tecnica* per ricavare le espressioni dell'algebra relazionale è quella di *procedere per passi*, ricordando che **i risultati di interrogazioni intermedie sono sempre relazioni**

Esempio 3 – 2\4

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

PASSO 1 : si cercano tutti gli impiegati che guadagnano più di 40 milioni

$\sigma_{\text{Stipendio} > 40}(\text{Impiegati})$

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Esempio 3 – 3\4

$\sigma_{Stipendio > 40}$ (Impiegati)

Matricola	Nome	Età	Stipendio
7309	Rossi	34	45
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

PASSO 2 : si associano tutti gli impiegati che guadagnano più di 40 milioni ai loro capi, sfruttando un equi-join con $Supervisione.Impiegato = Impiegati.Matricola$

Supervisione

Impiegato	Capo
7309	5698
5998	5698
9553	4076
5698	4076
4076	8123

Supervisione $\bowtie_{Impiegato=Matricola}$ $\sigma_{Stipendio > 40}$ (Impiegati)

Nome	Età	Stipendio	Impiegato	Capo
Rossi	34	45	7309	5698
Bruni	43	42	5698	4076
Mori	45	50	4076	8123

Esempio 3 – 4\4

Supervisione \bowtie **Impiegato=Matricola** $\sigma_{Stipendio>40}$ (**Impiegati**)

Nome	Età	Stipendio	Impiegato	Capo
Rossi	34	45	7309	5698
Bruni	43	42	5698	4076
Mori	45	50	4076	8123

PASSO 3 : proietto la relazione
ottenuta solo sull'attributo *Capo*

Π_{capo} (**Supervisione** \bowtie **Impiegato=Matricola**
 $\sigma_{Stipendio>40}$ (**Impiegati**))

Capo
5698
4076
8123

Esempio 4

Impiegati

Matricola	Nome	Età	Stipendio
-----------	------	-----	-----------

Supervisione

Impiegato	Capo
-----------	------

Trovare *nome* e *stipendio* dei capi degli impiegati che *guadagnano più di 40 milioni*

$\Pi_{nome, stipendio}(\$

Impiegato $\bowtie_{Matricola=Capo}$

$\Pi_{capo}(\text{Supervisione} \bowtie_{Impiegato=Matricola}$

$(\sigma_{Stipendio > 40}(\text{Impiegati})) \text{))}$

Nome	Stipendio
Bruni	42
Mori	50
Lupi	60

Esempio 5

Impiegati

Matricola	Nome	Età	Stipendio
-----------	------	-----	-----------

Supervisione

Impiegato	Capo
-----------	------

Si può sfruttare
l'operatore
rappresentante la
differenza...

Trovare le *matricole* dei capi i cui
impiegati guadagnano **tutti** più di 40
milioni.

$\Pi_{capo}(\text{Supervisione}) -$

$\Pi_{capo}(\text{Supervisione} \bowtie_{\text{Impiegato=Matricola}}$

$(\sigma_{\text{Stipendio} <= 40}(\text{Impiegati}))$)

Capo
8123

Esempio 6

Impiegati

Matricola	Nome	Età	Stipendio
-----------	------	-----	-----------

Supervisione

Impiegato	Capo
-----------	------

Trovare gli *impiegati* che guadagnano più del proprio capo, mostrando *matricola*, *nome* e *stipendio* dell'impiegato e del capo

$\Pi_{\text{Matricola, Nome, Stipendio, MatrC, NomeC, StipC}}($

$\sigma_{\text{Stipendio} > \text{StipC}}($

$\rho((\text{Matricola} \rightarrow \text{MatrC}, \text{Nome} \rightarrow \text{NomeC}, \text{Stipendio} \rightarrow \text{StipC}, \text{Età} \rightarrow \text{EtàC}), \text{Impiegati})$

$\bowtie_{\text{MatrC}=\text{Capo}}$

$(\text{Supervisione} \bowtie_{\text{Impiegato}=\text{Matricola}} \text{Impiegati})))$



MatrC	NomeC	StipC	Matricola	Nome	Stipendio
5698	Bruni	42	7309	Rossi	45

Esercizio 1

▶ **Si consideri il database che contiene le seguenti relazioni:**

- ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
- ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
- ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

▶ **Formulare in algebra relazionale le seguenti interrogazioni:**

- ▶ 1. Trovare i *nomi* dei fornitori che forniscono pezzi rossi
- ▶ 2. Trovare i *fid* dei fornitori che forniscono pezzi rossi o pezzi verdi
- ▶ 3. Trovare i *fid* dei fornitori che forniscono pezzi rossi o si trovano a via Cavour
- ▶ 4. Trovare i *fid* dei fornitori che forniscono pezzi rossi e pezzi verdi
- ▶ 5. Trovare i *fid* dei fornitori che forniscono tutti i pezzi
- ▶ 6. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi
- ▶ 7. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi o verdi
- ▶ 8. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi oppure tutti i pezzi verdi
- ▶ 9. Trovare coppie di *fid* tali che il fornitore con il primo *fid* applica per alcune parti un prezzo maggiore di quello del fornitore con il secondo *fid*
- ▶ 10. Trovare i *pid* dei pezzi forniti da almeno due diversi fornitori
- ▶ 11. Trovare i *pid* dei pezzi più costosi forniti dai fornitori chiamati “Sapienza”
- ▶ 12. Trovare i *pid* dei pezzi forniti da tutti i fornitori a meno di 200 € (se qualche fornitore non ne fornisce, o chiede più di 200 €, il pezzo non viene selezionato)

Soluzione Esercizio 1 - 1\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

1. Trovare i *nomi* dei fornitori che forniscono pezzi rossi

$\pi_{fnome} ($
 $\pi_{fid}((\pi_{pid}(\sigma_{colore='rosso'}(Pezzi))) \bowtie Catalogo)$
 $\bowtie Fornitori)$

Le proiezioni π_{fid} e π_{pid} sono **ridondanti**...migliorano la “leggibilità” dei risultati intermedi, ma in questo caso non sono strettamente necessarie

Soluzione Esercizio 1 - 2\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fname* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)
-

2. Trovare i *fid* dei fornitori che forniscono pezzi rossi o pezzi verdi

$$\pi_{fid}(\pi_{pid}(\sigma_{colore='rosso' \cup colore='verde'}(Pezzi)) \bowtie Catalogo)$$

Soluzione Esercizio 1 - 3\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

3. Trovare i *fid* dei fornitori che forniscono pezzi rossi o si trovano a via Cavour

$\rho(R1, \pi_{fid}((\pi_{pid}(\sigma_{colore='rosso'}(Pezzi))) \bowtie Catalogo))$

$\rho(R2, \pi_{fid}(\sigma_{indirizzo='Via Cavour'}(Fornitori)))$

$R1 \cup R2$

Si sfrutta l'operatore di ridenominazione per creare le due nuove relazioni R1 e R2, sulle quali si può successivamente calcolare l'unione

Soluzione Esercizio 1 - 4\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

4. Trovare i *fid* dei fornitori che forniscono pezzi rossi e pezzi verdi

$$\rho(R1, \pi_{fid}(\pi_{pid}(\sigma_{colore='rosso'}(Pezzi)) \bowtie Catalogo))$$
$$\rho(R2, \pi_{fid}(\pi_{pid}(\sigma_{colore='verde'}(Pezzi)) \bowtie Catalogo))$$
$$R1 \cap R2$$

Soluzione Esercizio 1 - 5\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

5. Trovare i *fid* dei fornitori che forniscono tutti i pezzi

$$(\pi_{fid, pid} (Catalogo)) / (\pi_{pid} (Pezzi))$$



Si sfrutta l'operatore di divisione

Soluzione Esercizio 1 - 6\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)
-

6. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi

$$(\pi_{fid, pid} (Catalogo) / (\pi_{pid} (\sigma_{colore='rosso'} (Pezzi)))$$

Soluzione Esercizio 1 - 7\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)
-

7. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi o verdi

$$(\pi_{fid, pid} (Catalogo)) / (\pi_{pid} (\sigma_{colore='rosso' \cup colore='verde'} (Pezzi)))$$

Soluzione Esercizio 1 - 8\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

8. Trovare i *fid* dei fornitori che forniscono tutti i pezzi rossi oppure tutti i pezzi verdi

$$\rho(R1, ((\pi_{fid, pid} (Catalogo)) / (\pi_{pid} (\sigma_{colore='rosso'} (Pezzi))))$$

$$\rho(R2, ((\pi_{fid, pid} (Catalogo)) / (\pi_{pid} (\sigma_{colore='verde'} (Pezzi))))$$

$$R1 \cup R2$$

Soluzione Esercizio 1 - 9\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

9. Trovare coppie di *fid* tali che il fornitore con il primo *fid* applica per alcuni pezzi un prezzo maggiore di quello del fornitore con il secondo *fid*

$\rho(R1, Catalogo)$

$\rho(R2, Catalogo)$

$\pi_{R1.fid, R2.fid}(\sigma_{R1.pid = R2.pid \wedge R1.fid \neq R2.fid \wedge R1.costo > R2.costo} (R1 \times R2))$

Soluzione Esercizio 1 - 10\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)
-

10. Trovare i *pid* dei pezzi forniti da almeno due diversi fornitori

$\rho(R1, Catalogo)$

$\rho(R2, Catalogo)$

$\pi_{R1.pid} (\sigma_{R1.pid = R2.pid \wedge R1.fid \neq R2.fid} (R1 \times R2))$

Soluzione Esercizio 1 - 11\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pname* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)

11. Trovare i *pid* dei pezzi più costosi forniti dai fornitori chiamati “Sapienza”

$\rho(R1, \pi_{fid}(\sigma_{fnome='Sapienza'}(Fornitori)))$

$\rho(R2, (R1 \bowtie Catalogo))$

$\rho(R3, R2)$

$\rho(R4(1 \rightarrow fid, 2 \rightarrow pid, 3 \rightarrow costo), (\sigma_{R3.costo < R2.costo}(R3 \times R2)))$

$\pi_{pid}(R2 - \pi_{fid, pid, costo}(R4))$

Per cercare i pezzi meno costosi si può cambiare segno alla disuguaglianza

La proiezione contiene tutti i prodotti che costano meno di almeno un altro

Soluzione Esercizio 1 - 12\12

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **FORNITORI** (*fid* : integer, *fnome* : String, *indirizzo* : String)
 - ▶ **PEZZI**(*pid* : integer , *pnome* : String, *colore* : String)
 - ▶ **CATALOGO** (*fid* : integer, *pid* : integer, *costo* : real)
-

12. Trovare i *pid* dei pezzi forniti da tutti i fornitori a meno di 200 € (se qualche fornitore non ne fornisce, o chiede più di 200 €, il pezzo non viene selezionato)

$$(\pi_{pid, fid} (\sigma_{costo \leq 200}(Catalogo))) / (\pi_{fid}(Fornitori))$$

Esercizio 2

Vincoli di integrità referenziale tra:

- *Commissioni.Presidente e Deputati.Codice*
- *Deputati.Commissione e Commissioni.Numero*
- *Deputati.Provincia e Province.Sigla*
- *Deputati.Collegio e Collegi.Numero*
- *Province.Regioni e Regioni.Codice*
- *Collegi.Provincia e Province.Sigla*

▶ **Si consideri il database che contiene le seguenti relazioni:**

- ▶ **DEPUTATI** (Codice, Cognome, Nome, Commissione, Provincia, Collegio)
- ▶ **COLLEGI** (Provincia, Numero, Nome)
- ▶ **PROVINCE** (Sigla, Nome, Regione)
- ▶ **REGIONI** (Codice, Nome)
- ▶ **COMMISSIONI** (Numero, Nome, Presidente)

▶ **Formulare in algebra relazionale le seguenti interrogazioni:**

- ▶ 1. Trovare nome e cognome dei presidenti di commissioni cui partecipa almeno un deputato eletto in una provincia della Sicilia
- ▶ 2. Trovare nome e cognome dei deputati della commissione Bilancio
- ▶ 3. Trovare nome, cognome e provincia di elezione dei deputati della commissione Bilancio
- ▶ 4. Trovare nome, cognome, provincia e regione di elezione dei deputati della commissione Bilancio
- ▶ 5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto
- ▶ 6. Trovare i collegi di una stessa regione in cui siano stati eletti deputati con lo stesso nome proprio

Soluzione Esercizio 2 - 1\6

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **DEPUTATI** (Codice, Cognome, Nome, Commissione, Provincia, Collegio)
 - ▶ **COLLEGI** (Provincia, Numero, Nome)
 - ▶ **PROVINCE** (Sigla, Nome, Regione)
 - ▶ **REGIONI** (Codice, Nome)
 - ▶ **COMMISSIONI** (Numero, Nome, Presidente)

1. Trovare nome e cognome dei presidenti di commissioni cui partecipa almeno un deputato eletto in una provincia della Sicilia

$\pi_{nom, cogn}$

$((\rho((nome \rightarrow nom, cognome \rightarrow cogn), Deputati) \bowtie_{Presidente=Codice}$

$(Commissioni \bowtie_{Numero=Commissione} (Deputati \bowtie_{Provincia=Sigla}$

$(Province \bowtie_{Regione=Codice}$

$\sigma_{Nome='Sicilia'}(Regioni))))))$

Soluzione Esercizio 2 - 2\6

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **DEPUTATI** (Codice, Cognome, Nome, Commissione, Provincia, Collegio)
 - ▶ **COLLEGI** (Provincia, Numero, Nome)
 - ▶ **PROVINCE** (Sigla, Nome, Regione)
 - ▶ **REGIONI** (Codice, Nome)
 - ▶ **COMMISSIONI** (Numero, Nome, Presidente)

2. Trovare nome e cognome dei deputati della commissione Bilancio

$\pi_{\text{NomeC, Cognome}} \left(\rho_{((\text{Nome} \rightarrow \text{NomeC}), \text{Deputati})} \bowtie_{\text{Commissione=Numero}} \left(\sigma_{\text{nome}='Bilancio'} (\text{Commissioni}) \right) \right)$

Soluzione Esercizio 2 - 3\6

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **DEPUTATI** (Codice, Cognome, Nome, Commissione, Provincia, Collegio)
 - ▶ **COLLEGI** (Provincia, Numero, Nome)
 - ▶ **PROVINCE** (Sigla, Nome, Regione)
 - ▶ **REGIONI** (Codice, Nome)
 - ▶ **COMMISSIONI** (Numero, Nome, Presidente)

3. Trovare nome, cognome e provincia di elezione dei deputati della commissione Bilancio

$$\begin{aligned} & \pi_{nomeC, Cognome, nom1}(\rho((Nome \rightarrow nom1), Province) \bowtie_{Sigla=Provincia} (\rho((Nome \rightarrow nomeC), Deputati) \bowtie_{Commissione=Numero} (\sigma_{nome='Bilancio'}(Commissioni)))) \end{aligned}$$

Soluzione Esercizio 2 - 4\6

- ▶ **Si consideri il database che contiene le seguenti relazioni:**
 - ▶ **DEPUTATI** (Codice, Cognome, Nome, Commissione, Provincia, Collegio)
 - ▶ **COLLEGI** (Provincia, Numero, Nome)
 - ▶ **PROVINCE** (Sigla, Nome, Regione)
 - ▶ **REGIONI** (Codice, Nome)
 - ▶ **COMMISSIONI** (Numero, Nome, Presidente)

4. Trovare nome, cognome, provincia e regione di elezione dei deputati della commissione Bilancio

$\pi_{nomeC, Cognome, nom1, reg}(\rho((Nome \rightarrow reg), Regioni) \bowtie_{Codice=Regione} \rho((Nome \rightarrow nom1), Province) \bowtie_{Sigla=Provincia} (\rho((Nome \rightarrow nomeC), Deputati) \bowtie_{Commissione=Numero} (\sigma_{nome='Bilancio'}(Commissioni))))$

$\rho((Nome \rightarrow reg), Regioni) \bowtie_{Codice=Regione}$

$\rho((Nome \rightarrow nom1), Province) \bowtie_{Sigla=Provincia}$

$(\rho((Nome \rightarrow nomeC), Deputati) \bowtie_{Commissione=Numero}$

$(\sigma_{nome='Bilancio'}(Commissioni)))$

Soluzione Esercizio 2 - 5\6

5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto

- ▶ **La soluzione può essere ottenuta procedendo per passi :**
- ▶ **1) Calcolare la lista completa delle regioni e dei rispettivi collegi associati**
- ▶ **2) Calcolare la lista completa delle regioni che hanno più di un collegio associato**
- ▶ **3) Effettuare la differenza insiemistica tra la relazione ottenuta al punto (1) e la relazione ottenuta la punto (2) ; la relazione risultato conterrà SOLO quelle regioni che hanno esattamente un unico collegio associato**
- ▶ **4) Per quelle regioni che hanno esattamente un solo collegio, indicare nome e cognome del deputato eletto**

Soluzione Esercizio 2 - 5\6

5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto

- ▶ **1) Calcolare la lista completa delle regioni e dei rispettivi collegi associati**

$\rho(R1,$

$(\pi_{\text{Regione}, \text{CodiceCollegio}}($

$\rho((\text{Nome} \rightarrow \text{NomeColl}, \text{Numero} \rightarrow \text{CodiceCollegio}), \text{Collegi}) \bowtie_{\text{Provincia}=\text{Sigla}}$

$(\text{Province} \bowtie_{\text{Regione}=\text{Codice}} \rho((\text{Nome} \rightarrow \text{NomeReg}), \text{Regioni))))$



R1

Regione	CodiceCollegio
---------	----------------

Per aumentare la leggibilità, **si rinomina con R1** il risultato dell'interrogazione

Soluzione Esercizio 2 - 5\6

5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto

- 2) Calcolare la lista completa delle regioni che hanno più di un collegio associato

$\rho(R2 (Regione \rightarrow Regione2, CodiceCollegio \rightarrow CodiceCollegio2), R1)$

Si costruisce una nuova relazione R2 identica ad R1, ridenominando gli attributi

$\rho(R3, (\pi_{Regione, CodiceCollegio}(R1 \bowtie_{Regione=Regione2 \text{ AND } CodiceCollegio \neq CodiceCollegio2} R2))))$

R3 conterrà tutte le regioni che hanno almeno due collegi differenti...quindi R3 **NON CONTERRA'** le regioni che hanno un solo collegio

Esempio

R1

Regione	CodiceCollegio
C_Lazio	C01
C_Lazio	C02
C_Toscana	C03

R2

Regione2	CodiceCollegio2
C_Lazio	C01
C_Lazio	C02
C_Toscana	C03

R3

Regione	CodiceCollegio
C_Lazio	C01
C_Lazio	C02

Soluzione Esercizio 2 - 5\6

5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto

- ▶ **3) Effettuare la differenza insiemistica tra la relazione ottenuta al punto (1) e la relazione ottenuta la punto (2) ; la relazione risultato conterrà SOLO quelle regioni che hanno esattamente un unico collegio associato**

$\rho(R4, R1 - R3)$

R4 conterrà tutte le regioni che hanno un solo collegio

Esempio

R1

Regione	CodiceCollegio
C_Lazio	C01
C_Lazio	C02
C_Toscana	C03

R3

Regione	CodiceCollegio
C_Lazio	C01
C_Lazio	C02

R4



Regione	CodiceCollegio
C_Toscana	C03

Soluzione Esercizio 2 - 5\6

5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto

- ▶ **4) Per quelle regioni che hanno esattamente un solo collegio, indicare nome e cognome del deputato eletto (JOIN con la relazione Deputati e PROJ sugli attributi *Nome, Cognome, Regione e Collegio* – quest'ultimo attributo facoltativo nella proiezione)**

$\rho(R5, (\pi_{Nome, Cognome, Regione, Collegio}($
 $Deputati \bowtie_{Collegio=CodiceCollegio} R4)))$



R5

Nome	Cognome	Regione	Collegio
------	---------	---------	----------

Soluzione Esercizio 2 - 6\6

6. Trovare i collegi di una stessa regione in cui siano stati eletti deputati con lo stesso nome proprio

1)

$\rho(R1,$

$\pi_{Regione, NomeReg, Collegio, NomeColl, Cognome, Nome($

$Deputati \bowtie_{Collegio=CodiceCollegio}$

$\rho((Nome \rightarrow NomeColl, Numero \rightarrow CodiceCollegio, Provincia \rightarrow ProvColl), Collegi \bowtie_{ProvColl=Sigla}$

$\rho((Nome \rightarrow NomeProv), Province \bowtie_{Regione=Codice}$

$\rho((Nome \rightarrow NomeReg), Regioni))))))$

R1 è la relazione che descrive la lista completa delle **Regioni**, dei **Collegi** associati e dei **Deputati** eletti

Soluzione Esercizio 2 - 6\6

6. Trovare i collegi di una stessa regione in cui siano stati eletti deputati con lo stesso nome proprio

2)

$\rho(R2(Regione \rightarrow Regione2, NomeReg \rightarrow NomeReg2, Collegio \rightarrow Collegio2, NomeColl \rightarrow NomeColl2, Nome \rightarrow Nome2, Cognome \rightarrow Cognome2), R1)$

Si costruisce una nuova relazione R2 identica ad R1, ridenominando gli attributi

3)

$\rho(R3, \pi_{NomeReg, NomeColl, Nome, Cognome}(R1 \bowtie_{Collegio \neq Collegio2 \text{ AND } Regione=Regione2 \text{ AND } Nome=Nome2} R2))$

Si effettua un JOIN condizionale fra R1 e R2, mantenendo quei deputati con lo stesso *Nome* eletti in diversi *Collegi della stessa Regione*

R3 fornisce tutti i collegi di una stessa regione in cui siano stati eletti deputati con lo stesso nome proprio. La soluzione ammette che uno stesso *Deputato* possa essere stato eletto per più collegi