

Selezione

- "seleziona" da una relazione le ennuple che soddisfano una condizione

Impiegati

Matricola	Cognome	Filiale	Stipendio
7309	Rossi	Roma	55
5998	Neri	Milano	64
9553	Milano	Milano	44
5698	Neri	Napoli	64

- impiegati che
 - guadagnano più di 50
 - guadagnano più di 50 e lavorano a Milano
 - hanno lo stesso nome della filiale presso cui lavorano

Selezione, sintassi e semantica

- Alla lavagna e su Relax
- Simbolo
 - SEL
 - σ

- impiegati che guadagnano più di 50

SEL_{Stipendio > 50} (Impiegati)

Matricola	Cognome	Filiale	Stipendio
7309	Rossi	Roma	55
5998	Neri	Milano	64
5698	Neri	Napoli	64

SEL_{Stipendio > 50} (Impiegati)

- impiegati che guadagnano più di 50 e lavorano a Milano

SEL_{Stipendio > 50 AND Filiale = 'Milano'} (Impiegati)

Matricola	Cognome	Filiale	Stipendio
5998	Neri	Milano	64

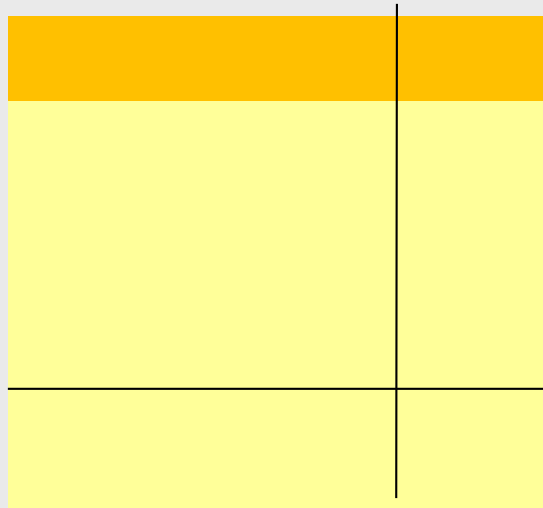
SEL_{Stipendio > 50 AND Filiale = 'Milano'} (Impiegati)

- impiegati che hanno lo stesso nome della filiale presso cui lavorano

SEL $\text{Cognome} = \text{Filiale}$ (**Impiegati**)

Matricola	Cognome	Filiale	Stipendio
9553	Milano	Milano	44

SEL $\text{Cognome} = \text{Filiale}$ (**Impiegati**)



selezione



Proiezione

- decompone "verticalmente":
 - "tutte" le ennuple, alcuni attributi

Impiegati

Matricola	Cognome	Filiale	Stipendio
7309	Neri	Napoli	55
5998	Neri	Milano	64
9553	Rossi	Roma	44
5698	Rossi	Roma	64

- per tutti gli impiegati:
 - matricola e cognome
 - cognome e filiale

Proiezione, sintassi e semantica

- alla lavagna

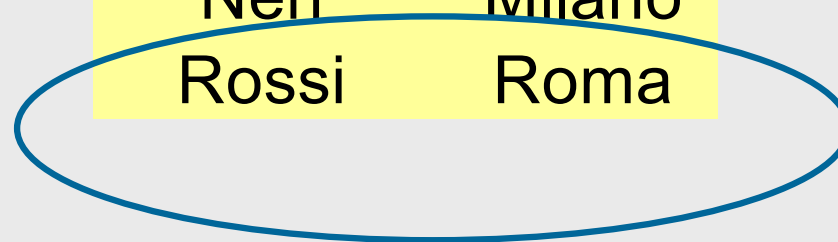
- matricola e cognome di tutti gli impiegati

Matricola	Cognome
7309	Neri
5998	Neri
9553	Rossi
5698	Rossi

PROJ Matricola, Cognome (Impiegati)

- cognome e filiale di tutti gli impiegati

Cognome	Filiale
Neri	Napoli
Neri	Milano
Rossi	Roma



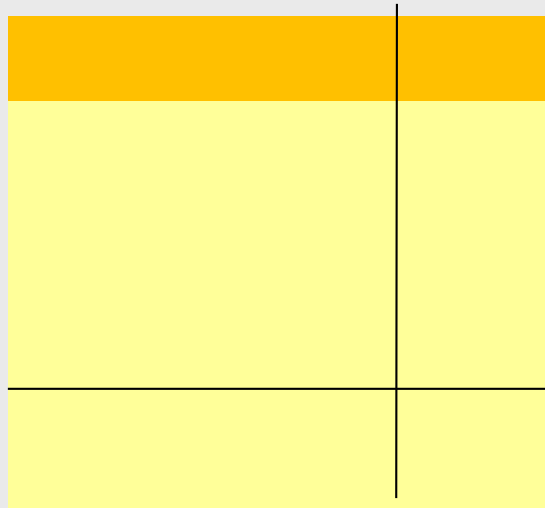
PROJ Cognome, Filiale (Impiegati)

Cardinalità delle proiezioni

- Una proiezione
 - contiene al più tante ennuple quante l'operando
 - può contenerne di meno
- Quando possiamo essere sicuri che non siano di meno?

Cardinalità delle proiezioni

- Proprietà interessante:
 - se X è una superchiave di R , allora $\text{PROJ}_X(R)$ contiene esattamente tante ennuple quante R



proiezione

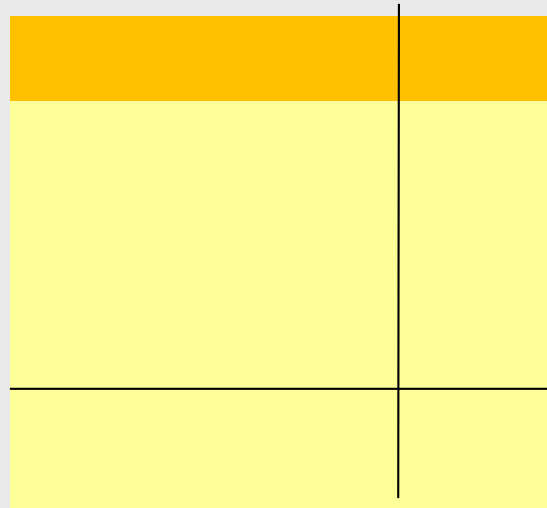


Selezione e proiezione

- operatori "ortogonali"
 - alla lavagna

Selezione e proiezione

- operatori "ortogonali"
- **selezione:**
 - decomposizione orizzontale
- **proiezione:**
 - decomposizione verticale



selezione



proiezione



Selezione con valori nulli

Impiegati

Matricola	Cognome	Filiale	Stipendio
7309	Rossi	Roma	32
5998	Neri	Milano	45
9553	Bruni	Milano	NULL

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

- la condizione atomica è vera solo per valori non nulli
- vediamo su Relax (relazione ImpiegatiTer)

<http://dbis-uibk.github.io/relax/calc/gist/1a9dc6cd0f3478388fc177dfc9b5a314>

Un risultato non desiderabile

$$\text{SEL}_{\text{Stipendio} > 40}(\text{Persone}) \cup \text{SEL}_{\text{Stipendio} \leq 40}(\text{Persone}) \neq \text{Persone}$$

- Vediamo su Relax.
- Perché?
 - Perché le selezioni vengono valutate separatamente!
- Ma anche

$$\text{SEL}_{\text{Stipendio} > 40 \text{ OR } \text{Stipendio} \leq 40}(\text{Persone}) \neq \text{Persone}$$

- Perché?
 - Perché anche le condizioni atomiche vengono valutate separatamente!

Selezione con valori nulli: soluzione

SEL Stipendio > 40 (Impiegati)

- la condizione atomica è vera solo per valori non nulli
- per riferirsi ai valori nulli esistono forme apposite di condizioni:

IS NULL

IS NOT NULL

- si potrebbe usare (ma non serve) una "logica a tre valori" (vero, falso, sconosciuto)

Nota bene

- RelaX usa

... = null

- Quindi:

$$\begin{aligned} & \text{SEL_Stipendio} > 40 \text{ (Persone)} \cup \text{SEL_Stipendio} \leq 40 \text{ (Persone)} \cup \\ & \quad \text{SEL_Stipendio IS NULL (Persone)} \\ & \quad = \\ & \text{SEL_Stipendio} > 40 \text{ OR Stipendio} \leq 40 \text{ OR Stipendio IS NULL (Persone)} \\ & \quad = \\ & \text{Persone} \end{aligned}$$

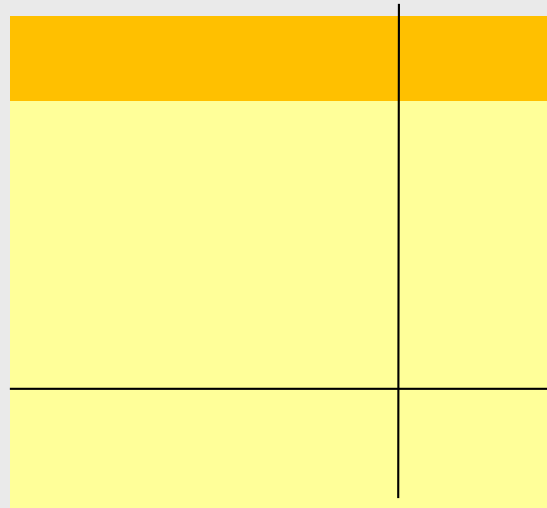
Impiegati

Matricola	Cognome	Filiale	Stipendio
5998	Neri	Milano	45
9553	Bruni	Milano	NULL

SEL (Stipendio > 40) OR (Stipendio IS NULL) (Impiegati)

Selezione e proiezione

- operatori "ortogonali"
- **selezione:**
 - decomposizione orizzontale
- **proiezione:**
 - decomposizione verticale



selezione



proiezione



Selezione e proiezione

- Combinando selezione e proiezione, possiamo estrarre interessanti informazioni da una relazione

- matricola e cognome degli impiegati che guadagnano più di 50

Matricola	Cognome
7309	Rossi
5998	Neri
5698	Neri

PROJ_{Matricola,Cognome} (SEL_{Stipendio > 50} (Impiegati))

- Combinando selezione e proiezione, possiamo **estrarre** informazioni da **una** relazione
- Invece non possiamo
 - calcolare informazioni derivate
 - correlare informazioni presenti in relazioni diverse, né informazioni in ennuple diverse di una stessa relazione
- Vediamo altri operatori che permettono queste attività

Join

- il join è l'operatore più interessante dell'algebra relazionale
- permette di correlare dati in relazioni diverse

Il solito esempio

(con nomi di attributi modificati in "esami")

studenti

<u>Matricola</u>	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1978
8765	Neri	Paolo	03/11/1976
9283	Verdi	Luisa	12/11/1979
3456	Rossi	Maria	01/02/1978

esami

<u>Matricola</u>	Voto	<u>Codice</u>
3456	30	04
3456	24	02
9283	28	01
6554	26	01

Che voti ha preso
Maria Rossi negli esami
che ha superato?

corsi

<u>Codice</u>	Titolo	Docente
01	Analisi	Mario
02	Chimica I	Bruni
04	Chimica II	Verdi

studenti

Matricola	Cognome	Nome	Data di nascita
3456	Rossi	Maria	01/02/1978

esami

Matricola	Voto	Codice
3456	30	04
3456	24	02

corsi

Codice	Titolo	Docente
02	Chimica I	Bruni
04	Chimica II	Verdi

(studenti JOIN esami) JOIN corsi

Matricola	Cognome	Nome	Data di nascita	Voto	Codice	Titolo	Docente
3456	Rossi	Maria	01/02/1978	30	04	Chimica II	Bruni
3456	Rossi	Maria	01/02/1978	24	02	Chimica I	Verdi

Join naturale

- operatore binario (generalizzabile)
- produce un risultato
 - sull'**unione** degli attributi degli operandi
 - con ennuple costruite ciascuna a partire da una ennupla di ognuno degli operandi

Join, sintassi e semantica

- alla lavagna

JOIN ~~×~~

JOIN NATURALE

$R_1(X_1) \quad R_2(X_2)$

$$R_1 \bowtie R_2 = \{ t \text{ su } X_1 X_2 \mid$$

e esistono $t_1 \in R_1$ e $t_2 \in R_2$

$$t[X_1] = t_1 \text{ e } t[X_2] = t_2 \}$$

Esempi

- alla lavagna
 - join completo, non completo, vuoto, $m \times n$

R_1

IMPIEGATO	REPARTO
Rossi	A
Neri	B
Bianchi	B

R_2

REPARTO	CAPO
A	Mori
B	Bruni

$R_1 \bowtie R_2$

IMPIEGATO	REPARTO	CAPO
Rossi	A	Mori
Neri	B	Bruni
Bianchi	B	Bruni

R_1

IMPIEGATO	REPARTO
Rossi	A
Neri	B
Bianchi	B

R_2

REPARTO	CAPO
A	Mori
B	Bruni

$R_1 \bowtie R_2$

IMPIEGATO REPARTO CAPO

R_1

IMPIEGATO	REPARTO
Rossi	A
Neri	B
Bianchi	B

R_2

REPARTO	CAPO
B	Mori
C	Bruni

$R_1 \bowtie R_2$

IMPIEGATO REPARTO CAPO

R_1

IMPIEGATO	REPARTO
Rossi	A
Neri	B
Bianchi	B

R_2

REPARTO	CAPD
C	Mori
D	Bruni

$R_1 \bowtie R_2$

IMPIEGATO REPARTO CAPD

R_1

IMPIEGATO	REPARTO
Rossi	A
Neri	A
Bianchi	A

R_2

REPARTO	CAPD
A	Mori
A	Bruni

$R_1 \bowtie R_2$

IMPIEGATO REPARTO CAPD

⋮

6 esempi!

R_1	A	B	C
1		1	1
2		1	2
3		2	2

R_2	B	C	D
1	1	1	1
2	1	3	
2	2	2	4

$R_1 \bowtie R_2$

A	B	C	D
---	---	---	---

Cardinalità del join

- Il join di R_1 e R_2 contiene un numero di ennuple ...
 - alla lavagna (chiavi, vincoli integrità referenziale)

R_1 $|R_1|$

R_2 $|R_2|$

$$0 \leq |R_1 \bowtie R_2| \leq |R_1| \cdot |R_2|$$

R_1

<u>A</u>	B
1	0
2	4
3	4

R_2

<u>B</u>	C
1	x
2	y
3	z
4	w
...	

$$0 \leq |R_1 \bowtie R_2| \leq |R_1|$$

R_1

<u>A</u>	B
1	1
2	4
3	4

R_2

<u>B</u>	C
1	x
2	y
3	z
4	w
...	

VINCOLO DI INTEGRITA' REF.

FRA $R_1(B)$ E LA CHIAVE B DI R_2

$$|R_1 \bowtie R_2| = |R_1|$$

Cardinalità del join

- Il join di R_1 e R_2 contiene un numero di ennuple
 - compreso fra zero e il prodotto di $|R_1|$ e $|R_2|$
- se il join coinvolge una chiave di R_2 , allora il numero di ennuple è
 - compreso fra zero e $|R_1|$
- se il join coinvolge una chiave di R_2 e un vincolo di integrità referenziale verso di essa, allora il numero di ennuple è
 - pari a $|R_1|$

Cardinalità del join, 2

- $R_1(A,B)$, $R_2(B,C)$
- in generale

$$0 \leq |R_1 \text{ JOIN } R_2| \leq |R_1| \times |R_2|$$

- se B è chiave in R_2

$$0 \leq |R_1 \text{ JOIN } R_2| \leq |R_1|$$

- se B è chiave in R_2 ed esiste vincolo di integrità referenziale fra B (in R_1) e R_2 :

$$|R_1 \text{ JOIN } R_2| = |R_1|$$