

# Lezione 3 – Algebra relazionale I

Prof.ssa Maria De Marsico  
demarsico@di.uniroma1.it



SAPIENZA  
UNIVERSITÀ DI ROMA

Linguaggio **formale** per interrogare una base di dati relazionale: consiste di un insieme di operatori che possono essere applicati a una (operatori *unari*) o due (operatori *binari*) istanze di relazione e forniscono come risultato una nuova istanza di relazione (che può essere «salvata» in una «variabile»)

Linguaggio **procedurale**: l'interrogazione consiste in un'espressione in cui compaiono operatori dell'algebra e istanze di relazioni della base di dati, in una sequenza che prescrive in maniera puntuale l'ordine delle operazioni e i loro operandi

- Consente di effettuare un “taglio verticale” su una relazione, cioè di selezionare solo alcune colonne (attributi)
- Si denota con il simbolo  $\pi$ :

- $\pi_{A1,A2,\dots,Ak}(r)$

- seleziona le colonne di  $r$  che corrispondono agli attributi  $A1, A2, \dots, Ak$

Cliente	Nome	C#	Città
	Rossi	C1	Roma
	Rossi	C2	Milano
	Bianchi	C3	Roma
	Verdi	C4	Roma

Query: Nomi dei clienti

$\pi_{\text{Nome}}(\text{Cliente})$

Rossi
Bianchi
Verdi

Cliente	Nome	C#	Città	.....
	Rossi	C1	Roma	.....
	Rossi	C2	Milano	.....
	Rossi	C3	Roma	.....
	Bianchi	C4	Roma	.....
	Verdi	C5	Roma	.....

Query: Nomi dei clienti

ATTENZIONE: Si seguono le regole insiemistiche  
Nella relazione risultato NON ci sono DUPLICATI

$\pi_{\text{Nome}}(\text{Cliente})$

Rossi  
Bianchi  
Verdi

DUPLICATO = tupla con TUTTI i valori ordinatamente uguali a quelli di un'altra tupla

Cliente	Nome	C#	Città	.....
	Rossi	C1	Roma	.....
	Rossi	C2	Milano	.....
	Rossi	C3	Roma	.....
	Bianchi	C4	Roma	.....
	Verdi	C5	Roma	.....

Proviamo ad  
aggiungere qualcosa

Query: Nomi, Città

$\pi_{\text{Nome,Città}}(\text{Cliente})$

Rossi	Roma
Rossi	Milano
Bianchi	Roma
Verdi	Milano

Meglio, ma ancora i DUPLICATI che vengono eliminati sono rilevanti

Cliente	Nome	C#	Città	.....
	Rossi	C1	Roma	.....
	Rossi	C2	Milano	.....
	Rossi	C3	Roma	.....
	Bianchi	C4	Roma	.....
	Verdi	C5	Roma	.....

Query: Nomi e codici  
dei clienti

Rossi	C1
Rossi	C2
Rossi	C3
Bianchi	C4
Verdi	C5

$\pi_{\text{Nome}, \text{C\#}}(\text{Cliente})$

Se vogliamo conservare i clienti **omonimi** dobbiamo aggiungere un ulteriore attributo, in questo caso la (una) «chiave» (il codice)

- Consente di effettuare un “taglio orizzontale” su una relazione, cioè di selezionare solo le righe (tuple) che soddisfano una data condizione
- Si denota con il simbolo  $\sigma$ :

- $\sigma_C(r)$

- seleziona le tuple di  $r$  che soddisfano la condizione  $C$



- La condizione di selezione è un'espressione **booleana composta** (tramite operatori  $\wedge, \vee$  e  $\neg$ ) in cui i termini semplici sono del tipo:
  - $A\theta B$
  - oppure
  - $A\theta'a'$
- dove:
- $\theta$  è un operatore di confronto ( $\theta \in \{<, =, >, \leq, \geq\}$ )
- A e B sono due attributi con lo stesso dominio ( $\text{dom}(A) = \text{dom}(B)$ )
- a è un elemento di  $\text{dom}(A)$  ( $a \in \text{dom}(A)$ ) (costante o espressione)

Cliente	Nome	C#	Città	.....
	Rossi	C1	Roma	.....
	Rossi	C2	Milano	.....
	Rossi	C3	Roma	.....
	Bianchi	C4	Roma	.....
	Verdi	C5	Roma	.....

Query: Dati dei clienti  
che risiedono a Roma

$\sigma_{\text{Città}='Roma'}(\text{Cliente})$

Rossi	C1	Roma	.....
Rossi	C3	Roma	.....
Bianchi	C4	Roma	.....
Verdi	C5	Roma	.....

Cliente	Nome	C#	Città	.....
	Rossi	C1	Roma	.....
	Rossi	C2	Milano	.....
	Rossi	C3	Roma	.....
	Bianchi	C4	Roma	.....
	Verdi	C5	Roma	.....

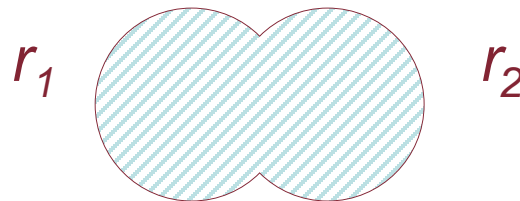
Query: Dati dei clienti  
che si chiamano Rossi  
e risiedono a Roma

$\sigma_{\text{Città}='Roma' \wedge \text{Nome}='Rossi'}(\text{Cliente})$

Rossi	C1	Roma	.....
Rossi	C3	Roma	.....

Con la sola selezione non si pone il problema della eventuale perdita di dati dovuta ai duplicati

- Consente di costruire una relazione contenente tutte le ennuple che appartengono ad almeno uno dei due operandi
- Si denota con il simbolo  $\cup$



- L'operazione di unione può essere applicata solo a operandi **union compatibili** cioè tali che:
  - hanno lo stesso numero di attributi
  - gli attributi corrispondenti (nell'ordine) sono definiti sullo stesso dominio
- **NOTA:** NON è necessario che gli attributi abbiano lo stesso nome, ma ovviamente il risultato ha senso se hanno un significato omogeneo
  - Es: anche se Numero di esami ed Età potrebbero essere definiti sullo stesso dominio, non avrebbe senso unire due relazioni che hanno questi attributi in colonne corrispondenti
  - Uno o entrambi gli operandi possono essere il risultato di operazioni precedenti, ad esempio per eliminare gli attributi incompatibili



## Docenti

Nome	CodDoc	Dipartimento
Rossi	C1	Matematica
Rossi	C2	Lettere
Bianchi	C3	Matematica
Verdi	C4	Lingue

## Amministrativi

Nome	CodAmm	Dipartimento
Esposito	C1	Lingue
Riccio	C2	Matematica
Pierro	C3	Lettere
Bianchi	C4	Lingue

Personale	Nome	Cod	Dipartimento
	Rossi	C1	Matematica
	Rossi	C2	Lettere
	Bianchi	C3	Matematica
	Verdi	C4	Lingue
	Esposito	C1	Lingue
	Riccio	C2	Matematica
	Pierro	C3	Lettere
	Bianchi	C4	Lingue

Personale = Docenti  $\cup$  Amministrativi

Potremmo avere problemi?



## Docenti

Nome	CodDoc	Dipartimento
Rossi	C1	Matematica
Rossi	C2	Lettere
Bianchi	C3	Matematica
Verdi	C4	Lingue

## Amministrativi

Nome	CodAmm	Dipartimento
Esposito	C1	Lingue
Riccio	C2	Matematica
Pierro	C3	Lettere
Verdi	C4	Lingue
Bianchi	C5	Lingue



Personale	Nome	Cod	Dipartimento
	Rossi	C1	Matematica
	Rossi	C2	Lettere
	Bianchi	C3	Matematica
	Verdi	C4	Lingue
	Esposito	C1	Lingue
	Riccio	C2	Matematica
	Pierro	C3	Lettere
	Bianchi	C5	Lingue

Personale = Docenti  $\cup$  Amministrativi  
Manca qualcuno?

## Docenti

Nome	CodDoc	Dipartimento
Rossi	D1	Matematica
Rossi	D2	Lettere
Bianchi	D3	Matematica
Verdi	D4	Lingue

## Amministrativi

Nome	CodAmm	Dipartimento
Esposito	A1	Lingue
Riccio	A2	Matematica
Pierro	A3	Lettere
Verdi	A4	Lingue
Bianchi	A5	Lingue



Personale	Nome	Cod	Dipartimento
	Rossi	D1	Matematica
	Rossi	D2	Lettere
	Bianchi	D3	Matematica
	Verdi	D4	Lingue
	Esposito	A1	Lingue
	Riccio	A2	Matematica
	Pierro	A3	Lettere
	Verdi	A4	Lingue
	Bianchi	A5	Lingue

Personale = Docenti  $\cup$  Amministrativi

- L'esempio precedente riguarda più regole pratiche di progettazione che le operazioni di algebra relazionale.
- Attenzione: il SQL il risultato di una query non segue la regola insiemistica  
MA
- Se ci sono due query coordinate unite da un operatore insiemistico UNION, la regola vale di nuovo

- Attenzione!

## Docenti

Nome	CodDoc	Dipartimento
Rossi	D1	Matematica
Rossi	D2	Lettere
Bianchi	D3	Matematica
Verdi	D4	Lingue

## Amministrativi

Nome	CodAmm	Dipartimento	Stip
Esposito	A1	Lingue	1250
Riccio	A2	Matematica	2000
Pierro	A3	Lettere	1000



- In questo esempio non è possibile unire le due relazioni in quanto **non sono union compatibili: hanno un numero diverso di attributi**



- Soluzione: proiettare su un sottoinsieme di attributi comuni (che abbiano un significato compatibile)
- Personale =
- $\pi_{\text{Nome, CodDoc, Dipartimento}}(\text{Docenti}) \cup$
- $\pi_{\text{Nome, CodAmm, Dipartimento}}(\text{Amministrativi})$

## Docenti

Nome	CodDoc	Dipartimento
Rossi	D1	Matematica
Rossi	D2	Lettere
Bianchi	D3	Matematica
Verdi	D4	Lingue

## Amministrativi

Nome	CodAmm	AnniServizio
Esposito	A1	10
Riccio	A2	15
Pierro	A3	2
Bianchi	A4	12



- In questo esempio non è possibile unire le due relazioni in quanto **non sono union compatibili**: attributi corrispondenti sono definiti su domini diversi

- Soluzione: proiettare su un sottoinsieme di attributi che abbiano lo stesso tipo (**e** che abbiano un significato compatibile)
- Personale =
- $\pi_{\text{Nome, CodDoc}}(\text{Docenti}) \cup$
- $\pi_{\text{Nome, CodAmm}}(\text{Amministrativi})$



## Docenti

Nome	CodDoc	Dipartimento
Rossi	D1	Matematica
Rossi	D2	Lettere
Bianchi	D3	Matematica
Verdi	D4	Lingue

## Amministrativi

Nome	CodAmm	Mansioni
Esposito	A1	Contabilità
Riccio	A2	Didattica
Pierro	A3	Segreteria
Bianchi	A4	Didattica

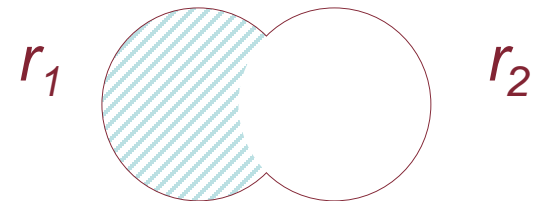


- In questo esempio le due relazioni **sono union compatibili** ma ... gli attributi anche se definiti sugli stessi domini hanno un significato DIVERSO

- Soluzione: proiettare su un sottoinsieme di attributi che abbiano lo stesso tipo **e** che abbiano un significato compatibile
- Personale =
- $\pi_{\text{Nome, CodDoc}}(\text{Docenti}) \cup$
- $\pi_{\text{Nome, CodAmm}}(\text{Amministrativi})$

- Si applica a operandi **union compatibili**
- Consente di costruire una relazione contenente tutte le tuple che appartengono al primo operando e NON appartengono al secondo operando

- Si denota con il simbolo –



- $r_1 - r_2$



## Studenti

Nome	CodFiscale	Dipartimento
Rossi	C1	Matematica
Rossi	C2	Lettere
Bianchi	C3	Matematica
Verdi	C4	Lingue

## Amministrativi

Nome	CodFiscale	Dipartimento
Esposito	C5	Lettere
Riccio	C6	Matematica
Pierro	C7	Lingue
Bianchi	C3	Matematica



- **Attenzione! La differenza non è commutativa come l'unione!**
- $\text{Studenti} - \text{Amministrativi} = \text{studenti che non sono anche amministrativi}$
- $\text{Amministrativi} - \text{Studenti} = \text{amministrativi che non sono anche studenti}$



## Studenti – Amministrativi

Studenti	Nome	CodFiscale	Dipartimento
	Rossi	C1	Matematica
	Rossi	C2	Lettere
	Verdi	C4	Lingue

## Amministrativi - Studenti

Amministrativi	Nome	CodFiscale	Dipartimento
	Esposito	C5	Lettere
	Riccio	C6	Matematica
	Pierro	C7	Lingue



- Valgono le stesse considerazioni dell'unione per relazioni che non sono union compatibili ma possono essere opportunamente proiettate per effettuare le operazioni di interesse

- Si applica a operandi **union compatibili**
- Consente di costruire una relazione contenente tutte le tuple che appartengono ad entrambi gli operandi

- Si denota con il simbolo  $\cap$  

- $r_1 \cap r_2 = (r_1 - (r_1 - r_2))$



## Studenti

Nome	CodFiscale	Dipartimento
Rossi	C1	Matematica
Rossi	C2	Lettere
Bianchi	C3	Matematica
Verdi	C4	Lingue

## Amministrativi

Nome	CodFiscale	Dipartimento
Esposito	C5	Lettere
Riccio	C6	Matematica
Pierro	C7	Lingue
Bianchi	C3	Matematica

- L'operazione di intersezione è commutativa
- $\text{Studenti} \cap \text{Amministrativi} = \text{studenti che sono anche amministrativi}$

**Studenti**  
**Amm**

Nome	CodFiscale	Dipartimento
Bianchi	C3	Matematica



- Valgono le stesse considerazioni dell'unione per relazioni che non sono union compatibili ma possono essere opportunamente proiettate per effettuare le operazioni di interesse.