



Modello relazionale e algebra relazionale

Algebra relazionale

Algebra relazionale

- Introduzione
- Selezione e proiezione
- Prodotto cartesiano e join
- Natural join, theta-join e semi-join
- Outer join
- Unione e intersezione
- Differenza e antijoin
- Divisione e altri operatori

Algebra relazionale

- Estende l'algebra degli insiemi per il modello relazionale
- Definisce un insieme di operatori che operano su relazioni e producono come risultato una relazione
- Gode della proprietà di chiusura
 - il risultato di qualunque operazione algebrica su relazioni è a sua volta una relazione

Operatori dell'algebra relazionale

➤ Operatori unari

- selezione (σ)
- proiezione (π)

➤ Operatori binari

- prodotto cartesiano (\times)
- join (\bowtie)
- unione (\cup)
- intersezione (\cap)
- differenza ($-$)
- divisione ($/$)

Operatori dell'algebra relazionale

➤ Operatori insiemistici

- unione (\cup)
- intersezione (\cap)
- differenza ($-$)
- prodotto cartesiano (\times)

➤ Operatori relazionali

- selezione (σ)
- proiezione (π)
- join (\bowtie)
- divisione ($/$)

Relazioni d'esempio

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Docenti

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica

- La selezione estrae un sottoinsieme *"orizzontale"* della relazione
- opera una decomposizione orizzontale della relazione

Selezione: esempio

➤ *Trovare i corsi tenuti nel secondo semestre*

Selezione: esempio

Corsi

Codice	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>
F1401	Elettronica	1	D104
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>



R

Codice	NomeCorso	Semestre	MatrDocente
M4880	Sistemi digitali	2	D104
F0410	Basi di dati	2	D102

Selezione: definizione

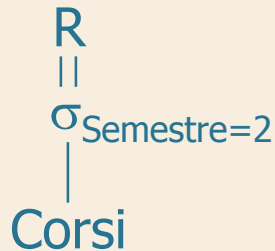
$$R = \sigma_p A$$

- La selezione genera una relazione R
 - avente lo stesso schema di A
 - contenente tutte le tuple della relazione A per cui è vero il predicato p
- Il predicato p è un'espressione booleana (operatori \wedge, \vee, \neg) di espressioni di confronto tra attributi o tra attributi e costanti
 - p : Città='Torino' \wedge Età>18
 - p : DataRestituzione>DataConsegna+10

Selezione: esempio

➤ *Trovare i corsi tenuti nel secondo semestre*

$$R = \sigma_{\text{Semestre}=2} \text{Corsi}$$



Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>
F1401	Elettronica	1	D104
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>

- La proiezione estrae un sottoinsieme *"verticale"* della relazione
- opera una decomposizione verticale della relazione



Proiezione: esempio (n. 1)

➤ *Trovare il nome dei docenti*

Proiezione: esempio (n. 1)

Docenti

<u>MatrDocente</u>	<i>NomeDoc</i>	Dipartimento
D102	<i>Verdi</i>	Informatica
D105	<i>Neri</i>	Informatica
D104	<i>Bianchi</i>	Elettronica



R

NomeDoc
Verdi
Neri
Bianchi

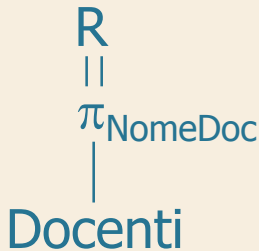
Proiezione: definizione

$$R = \pi_L A$$

- La proiezione genera una relazione R
- avente come schema la lista di attributi L (sottoinsieme dello schema di A)
 - contenente tutte le tuple presenti in A

Proiezione: esempio (n. 1)

➤ *Trovare il nome dei docenti*



$$R = \pi_{\text{NomeDoc}} \text{Docenti}$$

Docenti

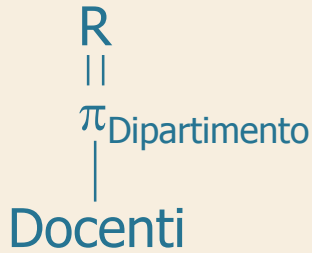
<u>MatrDocente</u>	<i>NomeDoc</i>	Dipartimento
D102	<i>Verdi</i>	Informatica
D105	<i>Neri</i>	Informatica
D104	<i>Bianchi</i>	Elettronica

Proiezione: esempio (n. 2)

➤ *Trovare i nomi dei dipartimenti in cui è presente almeno un docente*

Proiezione: esempio (n. 2)

➤ *Trovare i nomi dei dipartimenti in cui è presente almeno un docente*



$$R = \pi_{\text{Dipartimento}} \text{Docenti}$$

Proiezione: esempio (n. 2)

Docenti

<u>MatrDocente</u>	NomeDoc	<i>Dipartimento</i>
D102	Verdi	<i>Informatica</i>
D105	Neri	<i>Informatica</i>
D104	Bianchi	<i>Elettronica</i>



R

Dipartimento
Informatica
Elettronica

Proiezione: definizione

$$R = \pi_L A$$

- La proiezione genera una relazione R
 - avente come schema la lista di attributi L (sottoinsieme dello schema di A)
 - contenente tutte le tuple presenti in A
- Sono eliminati gli eventuali duplicati dovuti all'esclusione degli attributi non in L
 - se L include una chiave candidata, non vi sono duplicati

Selezione+proiezione: esempio

➤ *Selezionare il nome dei corsi nel secondo semestre*

Selezione+proiezione: esempio

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>
F1401	Elettronica	1	D104
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>



Selezione

Codice	NomeCorso	Semestre	MatrDocente
M4880	Sistemi digitali	2	D104
F0410	Basi di dati	2	D102

Selezione+proiezione: esempio

Codice	<i>NomeCorso</i>	Semestre	MatrDocente
M4880	<i>Sistemi digitali</i>	2	D104
F0410	<i>Basi di dati</i>	2	D102



Proiezione

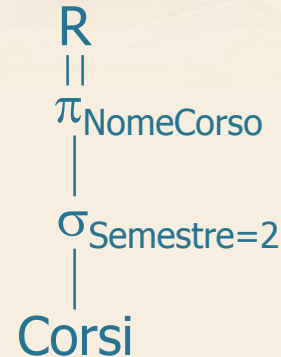
R

NomeCorso
Sistemi digitali
Basi di dati

Selezione+proiezione: esempio

➤ *Trovare il nome dei corsi nel secondo semestre*

$$R = \pi_{\text{NomeCorso}}(\sigma_{\text{Semestre}=2}\text{Corsi})$$



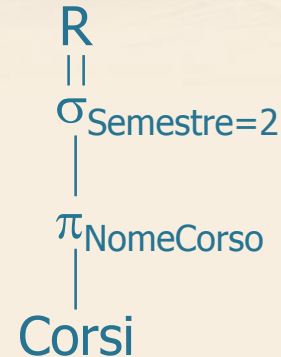
Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	<i>Sistemi digitali</i>	2	D104
F1401	Elettronica	1	D104
F0410	<i>Basi di dati</i>	2	D102

Selezione+proiezione: esempio (corretto?)

➤ *Trovare il nome dei corsi nel secondo semestre*

$$R = \sigma_{\text{Semestre}=2} (\pi_{\text{NomeCorso}} \text{Corsi})$$



Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Selezione+proiezione: soluzione errata

Corsi

<u>Codice</u>	<i>NomeCorso</i>	Semestre	MatrDocente
M2170	<i>Informatica 1</i>	1	D102
M4880	<i>Sistemi digitali</i>	2	D104
F1401	<i>Elettronica</i>	1	D104
F0410	<i>Basi di dati</i>	2	D102



Proiezione

NomeCorso
Informatica 1
Sistemi digitali
Elettronica
Basi di dati

Selezione+proiezione: soluzione errata

NomeCorso
Informatica 1
Sistemi digitali
Elettronica
Basi di dati

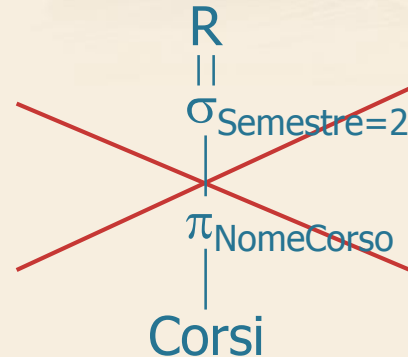
⇒ L'attributo Semestre non esiste più

- non è più disponibile l'informazione relativa al semestre
- non si può eseguire l'operazione di selezione

Selezione+proiezione: soluzione errata

➤ *Trovare il nome dei corsi nel secondo semestre*

~~$R = \sigma_{\text{Semestre}=2} (\pi_{\text{NomeCorso}} \text{Corsi})$~~



Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Prodotto cartesiano

- Il prodotto cartesiano di due relazioni A e B genera tutte le coppie formate da una tupla di A e una tupla di B

Prodotto cartesiano: esempio

➤ *Trovare il prodotto cartesiano tra Corsi e Docenti*

Prodotto cartesiano: esempio

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Docenti

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica

Prodotto cartesiano: esempio

R

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente	Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
M2170	Informatica 1	1	D102	D102	Verdi	Informatica
M2170	Informatica 1	1	D102	D105	Neri	Informatica
M2170	Informatica 1	1	D102	D104	Bianchi	Elettronica
M4880	Sistemi digitali	2	D104	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	D105	Neri	Informatica
M4880	Sistemi digitali	2	D104	D104	Bianchi	Elettronica
<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>	<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>	<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>	<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>	<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>	<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>	<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>

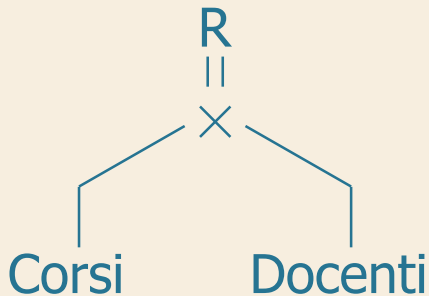
Prodotto cartesiano: definizione

$$R = A \times B$$

- Il prodotto cartesiano di due relazioni A e B genera una relazione R
- avente come schema l'unione degli schemi di A e di B
 - contenente tutte le coppie formate da una tupla di A e una tupla di B
- Il prodotto cartesiano è
- commutativo
 - $A \times B = B \times A$
 - associativo
 - $(A \times B) \times C = A \times (B \times C)$

Prodotto cartesiano: esempio

➤ *Trovare il prodotto cartesiano tra Corsi e Docenti*



$$R = \text{Corsi} \times \text{Docenti}$$

Legame tra attributi

R

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente	Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
M2170	Informatica 1	1	D102	D102	Verdi	Informatica
M2170	Informatica 1	1	D102	D105	Neri	Informatica
M2170	Informatica 1	1	D102	D104	Bianchi	Elettronica
M4880	Sistemi digitali	2	D104	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	D105	Neri	Informatica
M4880	Sistemi digitali	2	D104	D104	Bianchi	Elettronica
F1401	Elettronica	1	D104	D102	Verdi	Informatica
F1401	Elettronica	1	D104	D105	Neri	Informatica
F1401	Elettronica	1	D104	D104	Bianchi	Elettronica
F0410	Basi di dati	2	D102	D102	Verdi	Informatica
F0410	Basi di dati	2	D102	D105	Neri	Informatica
F0410	Basi di dati	2	D102	D104	Bianchi	Elettronica

- Il join di due relazioni A e B genera tutte le coppie formate da una tupla di A e una tupla di B *"semanticamente legate"*

Join: esempio

➤ *Trovare le informazioni sui corsi e sui docenti che li tengono*

Join: esempio

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente	Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
<i>M2170</i>	<i>Informatica 1</i>	<i>1</i>	<i>D102</i>	<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
M2170	Informatica 1	1	D102	D105	Neri	Informatica
M2170	Informatica 1	1	D102	D104	Bianchi	Elettronica
M4880	Sistemi digitali	2	D104	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	D105	Neri	Informatica
<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>	<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>
F1401	Elettronica	1	D104	D102	Verdi	Informatica
F1401	Elettronica	1	D104	D105	Neri	Informatica
<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>	<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>	<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
F0410	Basi di dati	2	D102	D105	Neri	Informatica
F0410	Basi di dati	2	D102	D104	Bianchi	Elettronica

Join: esempio

R

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente	Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
M2170	Informatica 1	1	D102	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	D104	Bianchi	Elettronica
F1401	Elettronica	1	D104	D104	Bianchi	Elettronica
F0410	Basi di dati	2	D102	D102	Verdi	Informatica

⇒ *Nota bene:* il docente (D105,Neri,Informatica), che non tiene alcun corso, non compare nel risultato del join

Join: definizione

- Il join è un operatore derivato
 - può essere espresso utilizzando gli operatori \times , σ_p , π_L
- Il join è definito separatamente perché esprime sinteticamente molte operazioni ricorrenti nelle interrogazioni
- Esistono diversi tipi di join
 - natural join
 - theta-join (e il suo sottocaso equi-join)
 - semi-join

Natural join: definizione

$$R = A \bowtie B$$

➤ Il natural join di due relazioni A e B genera una relazione R

- **avente come schema**
 - gli attributi presenti nello schema di A e non presenti nello schema di B
 - gli attributi presenti nello schema di B e non presenti nello schema di A
 - una sola copia degli attributi comuni (con lo stesso nome nello schema di A e di B)
- **contenente tutte le coppie costituite da una tupla di A e una tupla di B per cui il valore degli attributi comuni è uguale**

Natural join: proprietà

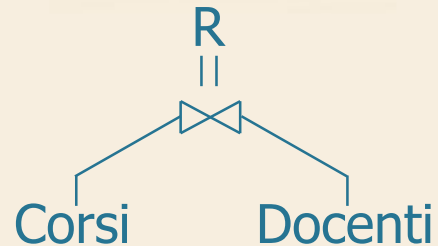
$$R = A \bowtie B$$

⇒ Il natural join è commutativo e associativo

Natural join: esempio

➤ *Trovare le informazioni sui corsi e sui docenti che li tengono*

$R = \text{Corsi} \bowtie \text{Docenti}$



R

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
M2170	Informatica 1	1	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	Bianchi	Elettronica
F1401	Elettronica	1	D104	Bianchi	Elettronica
F0410	Basi di dati	2	D102	Verdi	Informatica

Natural join: esempio

R

Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
M2170	Informatica 1	1	D102	Verdi	Informatica
M4880	Sistemi digitali	2	D104	Bianchi	Elettronica
F1401	Elettronica	1	D104	Bianchi	Elettronica
F0410	Basi di dati	2	D102	Verdi	Informatica

➤ *Nota bene:* l'attributo comune MatrDocente è presente una volta sola nello schema della relazione risultante R

Theta-join

- Il theta-join di due relazioni A e B genera tutte le coppie formate da una tupla di A e una tupla di B che soddisfano una generica "*condizione di legame*"

Theta-join: esempio

➤ *Trovare la matricola dei docenti che sono titolari di almeno due corsi*

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Theta-join: esempio

Corsi C1

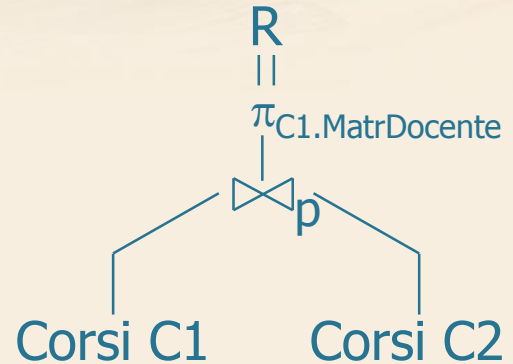
<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Corsi C2

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

Theta-join: esempio

➤ *Trovare la matricola dei docenti che sono titolari di almeno due corsi*



$p: C1.MatrDocente=C2.MatrDocente \wedge C1.Codice \neq C2.Codice$

$R = \pi_{C1.MatrDocente}((Corsi\ C1) \bowtie_p (Corsi\ C2))$

Theta-join: esempio

Corsi C1. Codice	Corsi C1. NomeCorso	Corsi C1. Semestre	Corsi C1. MatrDocente	Corsi C2. Codice	Corsi C2. NomeCorso	Corsi C2. Semestre	Corsi C2. MatrDocente
M2170	Informatica 1	1	D102	M2170	Informatica 1	1	D102
M2170	Informatica 1	1	D102	M4880	Sistemi digitali	2	D104
M2170	Informatica 1	1	D102	F1401	Elettronica	1	D104
<i>M2170</i>	<i>Informatica 1</i>	<i>1</i>	<i>D102</i>	<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>
M4880	Sistemi digitali	2	D104	M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104	M4880	Sistemi digitali	2	D104
<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>	<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>
M4880	Sistemi digitali	2	D104	F0410	Basi di dati	2	D102
F1401	Elettronica	1	D104	M2170	Informatica 1	1	D102
<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>	<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>
F1401	Elettronica	1	D104	F1401	Elettronica	1	D104
F1401	Elettronica	1	D104	F0410	Basi di dati	2	D102
<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>	<i>M2170</i>	<i>Informatica 1</i>	<i>1</i>	<i>D102</i>
F0410	Basi di dati	2	D102	M4880	Sistemi digitali	2	D104
F0410	Basi di dati	2	D102	F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102	F0410	Basi di dati	2	D102

Theta-join: esempio

Corsi C1. Codice	Corsi C1. NomeCorso	Corsi C1. Semestre	Corsi C1. MatrDocente	Corsi C2. Codice	Corsi C2. NomeCorso	Corsi C2. Semestre	Corsi C2. MatrDocente
M2170	Informatica 1	1	D102	F0410	Basi di dati	2	D102
M4880	Sistemi digitali	2	D104	F1401	Elettronica	1	D104
F1401	Elettronica	1	D104	M4880	Sistemi digitali	2	D104
F0410	Basi di dati	2	D102	M2170	Informatica 1	1	D102

Theta-join: esempio

Corsi C1. Codice	Corsi C1. NomeCorso	Corsi C1. Semestre	Corsi C1. MatrDocente	Corsi C2. Codice	Corsi C2. NomeCorso	Corsi C2. Semestre	Corsi C2. MatrDocente
M2170	Informatica 1	1	<i>D102</i>	F0410	Basi di dati	2	D102
M4880	Sistemi digitali	2	<i>D104</i>	F1401	Elettronica	1	D104
F1401	Elettronica	1	<i>D104</i>	M4880	Sistemi digitali	2	D104
F0410	Basi di dati	2	<i>D102</i>	M2170	Informatica 1	1	D102



R

Corsi C1. MatrDocente
D102
D104

Theta-join: definizione

$$R = A \bowtie_p B$$

- Il theta-join di due relazioni A e B genera una relazione R
 - avente come schema l'unione degli schemi di A e di B
 - contenente tutte le coppie costituite da una tupla di A e una tupla di B per cui è vero il predicato p
- Il predicato p è nella forma $X \theta Y$
 - X è un attributo di A, Y è un attributo di B
 - θ è un operatore di confronto compatibile con i domini di X e di Y
- Il theta-join è commutativo e associativo

Equi-join: definizione

$$R = A \bowtie_p B$$

➤ Equi-join

- caso particolare del theta-join in cui θ è l'operatore di uguaglianza (=)

Semi-join

- Il semi-join di due relazioni A e B seleziona tutte le tuple di A "*semanticamente legate*" ad almeno una tupla di B
- le informazioni di B non compaiono nel risultato

Semi-join: esempio

➤ *Trovare le informazioni relative ai docenti titolari di almeno un corso*

Semi-join: esempio

Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento	Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>	<i>M2170</i>	<i>Informatica 1</i>	<i>1</i>	<i>D102</i>
D102	Verdi	Informatica	M4880	Sistemi digitali	2	D104
D102	Verdi	Informatica	F1401	Elettronica	1	D104
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>	<i>F0410</i>	<i>Basi di dati</i>	<i>2</i>	<i>D102</i>
D105	Neri	Informatica	M2170	Informatica 1	1	D102
D105	Neri	Informatica	M4880	Sistemi digitali	2	D104
D105	Neri	Informatica	F1401	Elettronica	1	D104
D105	Neri	Informatica	F0410	Basi di dati	2	D102
D104	Bianchi	Elettronica	M2170	Informatica 1	1	D102
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>	<i>M4880</i>	<i>Sistemi digitali</i>	<i>2</i>	<i>D104</i>
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>	<i>F1401</i>	<i>Elettronica</i>	<i>1</i>	<i>D104</i>
D104	Bianchi	Elettronica	F0410	Basi di dati	2	D102

Semi-join: esempio

<i>Docenti. MatrDocente</i>	<i>Docenti. NomeDoc</i>	<i>Docenti. Dipartimento</i>	Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>	M2170	Informatica 1	1	D102
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>	F0410	Basi di dati	2	D102
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>	M4880	Sistemi digitali	2	D104
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>	F1401	Elettronica	3	D104



R

Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
D102	Verdi	Informatica
D104	Bianchi	Elettronica

Semi-join: definizione

$$R = A \bowtie_p B$$

- Il semi-join di due relazioni A e B genera una relazione R
 - avente lo stesso schema di A
 - contenente tutte le tuple di A per cui è vero il predicato specificato da p
- Il predicato p è espresso nella stessa forma del theta-join (confronto tra attributi di A e di B)

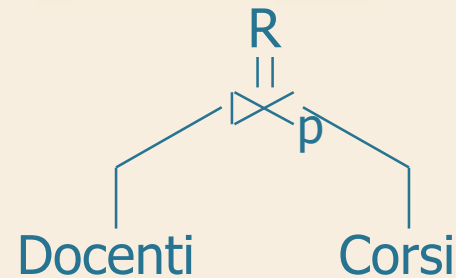
Semi-join: proprietà

- Il semi-join può essere espresso in funzione del theta-join
 - $A \bowtie_p B = \pi_{\text{schema}(A)}(A \bowtie_p B)$
- Il semi-join *non gode* della proprietà commutativa

Semi-join: esempio

➤ *Trovare le informazioni relative ai docenti titolari di almeno un corso*

$R = \text{Docenti} \bowtie_p \text{Corsi}$



$p: \text{Docenti.MatrDocente} = \text{Corsi.MatrDocente}$

R

Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento
D102	Verdi	Informatica
D104	Bianchi	Elettronica

- Variante del join che permette di conservare l'informazione relativa alle tuple non semanticamente legate dal predicato di join
 - completa con valori nulli le tuple prive di controparte
- Esistono tre tipi di outer-join
 - left: sono completate solo le tuple del primo operando
 - right: sono completate solo le tuple del secondo operando
 - full: sono completate le tuple di entrambi gli operandi

Left outer-join

- Il left outer-join di due relazioni A e B genera le coppie formate da
- una tupla di A e una di B "*semanticamente legate*"
- +
- una tupla di A "*non semanticamente legata*" a tuple di B completata con valori nulli per tutti gli attributi di B

Left outer-join: esempio

➤ *Trovare le informazioni sui docenti e sui corsi che tengono*

Left outer-join: esempio

R

Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento	Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente
D102	Verdi	Informatica	M2170	Informatica 1	1	D102
D102	Verdi	Informatica	F0410	Basi di dati	2	D102
D104	Bianchi	Elettronica	M4880	Sistemi digitali	2	D104
D104	Bianchi	Elettronica	F1401	Elettronica	1	D104
<i>D105</i>	<i>Neri</i>	<i>Informatica</i>	<i>null</i>	<i>null</i>	<i>null</i>	<i>null</i>

Left outer-join: definizione

$$R = A \bowtie_p B$$

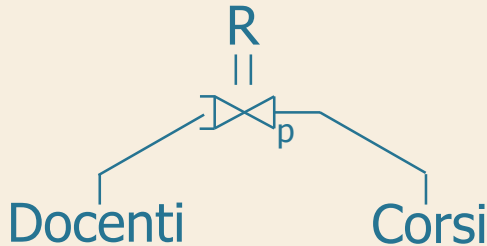
➤ Il left outer-join di due relazioni A e B genera una relazione R

- avente come schema l'unione degli schemi di A e di B
- contenente le coppie formate da
 - una tupla di A e una tupla di B per cui è vero il predicato p
 - una tupla di A che non è correlata mediante il predicato p a tuple di B completata con valori nulli per tutti gli attributi di B

➤ Il left outer-join *non è* commutativo

Left outer-join: esempio

➤ *Trovare le informazioni sui docenti e sui corsi che tengono*



$R = \text{Docenti} \bowtie_p \text{Corsi}$

R p: Docenti.MatrDocente=Corsi.MatrDocente

Docenti. MatrDocente	Docenti. NomeDoc	Docenti. Dipartimento	Corsi. Codice	Corsi. NomeCorso	Corsi. Semestre	Corsi. MatrDocente
D102	Verdi	Informatica	M2170	Informatica 1	1	D102
D102	Verdi	Informatica	F0410	Basi di dati	2	D102
D104	Bianchi	Elettronica	M4880	Sistemi digitali	2	D104
D104	Bianchi	Elettronica	F1401	Elettronica	1	D104
D105	Neri	Informatica	null	null	null	null

Right outer-join: definizione

$$R = A \bowtie_p B$$

➤ Il right outer-join di due relazioni A e B genera una relazione R

- avente come schema l'unione degli schemi di A e di B
- contenente le coppie formate da
 - una tupla di A e una tupla di B per cui è vero il predicato p
 - una tupla di B che non è correlata mediante il predicato p a tuple di A completata con valori nulli per tutti gli attributi di A

Il right outer-join *non è* commutativo

Full outer-join: definizione

$$R = A \bowtie_p B$$

- Il full outer-join di due relazioni A e B genera una relazione R
- avente come schema l'unione degli schemi di A e di B

Full outer-join: definizione

$$R = A \bowtie_p B$$

➤ Il full outer-join di due relazioni A e B genera una relazione R

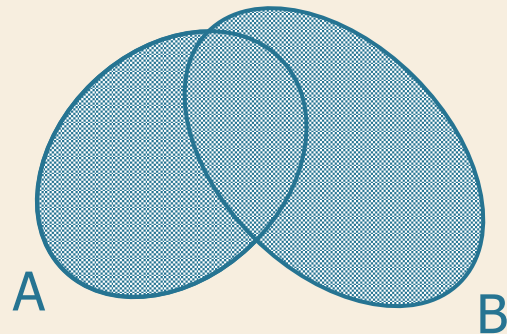
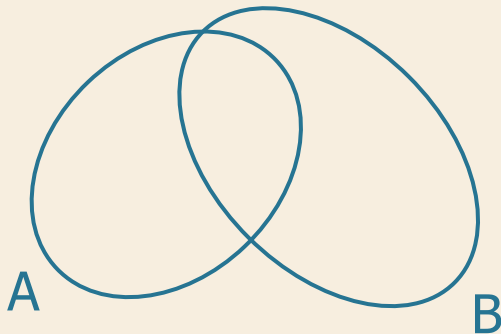
- contenente le coppie formate da
 - una tupla di A e una tupla di B per cui è vero il predicato p
 - una tupla di A che non è correlata mediante il predicato p a tuple di B completata con valori nulli per tutti gli attributi di B
 - una tupla di B che non è correlata mediante il predicato p a tuple di A completata con valori nulli per tutti gli attributi di A

Full outer-join: proprietà

$$R = A \bowtie_p B$$

⇒ Il full outer-join è commutativo

➤ L'unione di due relazioni A e B seleziona tutte le tuple presenti in almeno una delle due relazioni



Unione: relazioni d'esempio

DocentiLaurea

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica

DocentiMaster

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
D101	Rossi	Elettrica

Unione: esempio

➤ *Trovare le informazioni relative ai docenti dei corsi di laurea o di master*

Unione: esempio

DocentiLaurea

MatrDocente	NomeDoc	Dipartimento
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>

DocentiMaster

MatrDocente	NomeDoc	Dipartimento
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
<i>D101</i>	<i>Rossi</i>	<i>Elettrica</i>

R

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica
D101	Rossi	Elettrica

➤ *Nota bene:*
i duplicati sono eliminati

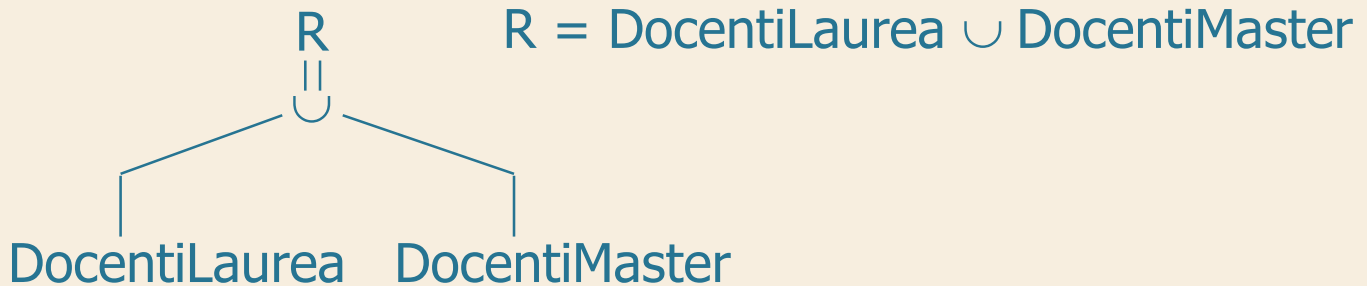
Unione: definizione

$$R = A \cup B$$

- L'unione di due relazioni A e B genera una relazione R
 - avente lo stesso schema di A e B
 - contenente tutte le tuple appartenenti ad A e tutte le tuple appartenenti a B (o a entrambi)
- *Compatibilità*
 - le relazioni A e B devono avere lo stesso schema (numero e tipo degli attributi)
- Le tuple duplicate sono eliminate
- L'unione è commutativa e associativa

Unione: esempio

➤ *Trovare le informazioni relative ai docenti dei corsi di laurea o di master*

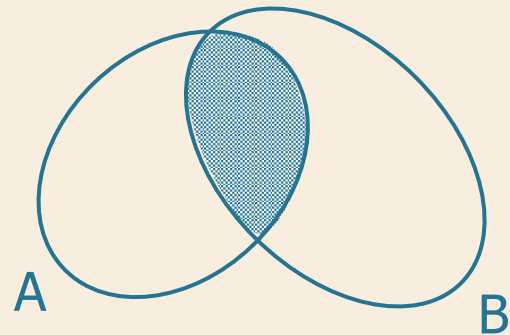
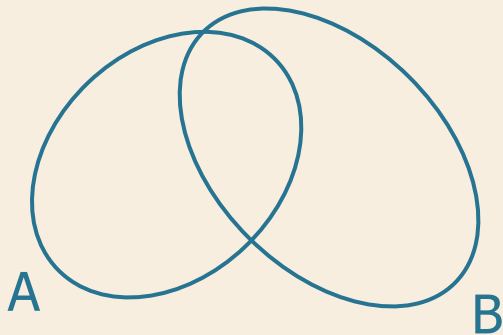


R

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica
D101	Rossi	Elettrica

Intersezione

➤ L'intersezione di due relazioni A e B seleziona tutte le tuple presenti in entrambe le relazioni



Intersezione: esempio

- *Trovare le informazioni relative ai docenti sia di corsi di laurea, sia di master*

Intersezione: esempio

DocentiLaurea

MatrDocente	NomeDoc	Dipartimento
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
D105	Neri	Informatica
D104	Bianchi	Elettronica

DocentiMaster

MatrDocente	NomeDoc	Dipartimento
<i>D102</i>	<i>Verdi</i>	<i>Informatica</i>
D101	Rossi	Elettrica



R

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica

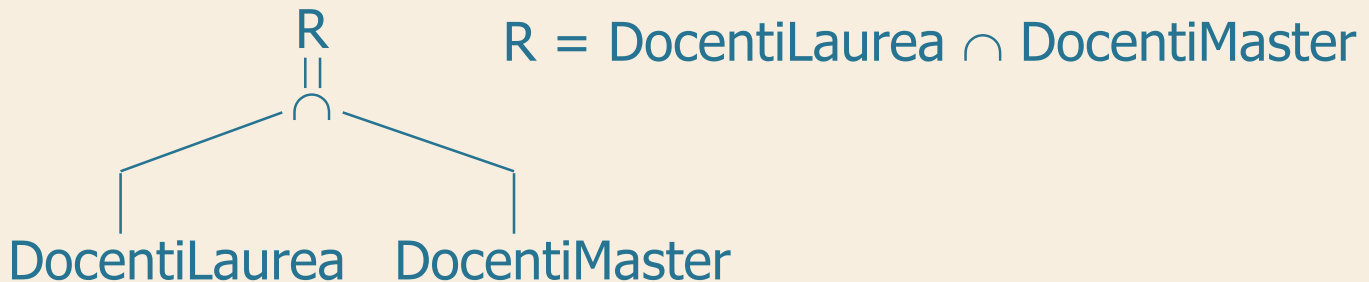
Intersezione: definizione

$$R = A \cap B$$

- L'intersezione di due relazioni A e B genera una relazione R
 - avente lo stesso schema di A e B
 - contenente tutte le tuple appartenenti sia ad A sia a B
- *Compatibilità*
 - le relazioni A e B devono avere lo stesso schema (numero e tipo degli attributi)
- L'intersezione è commutativa e associativa

Intersezione: esempio

➤ *Trovare le informazioni relative ai docenti sia di corsi di laurea, sia di master*

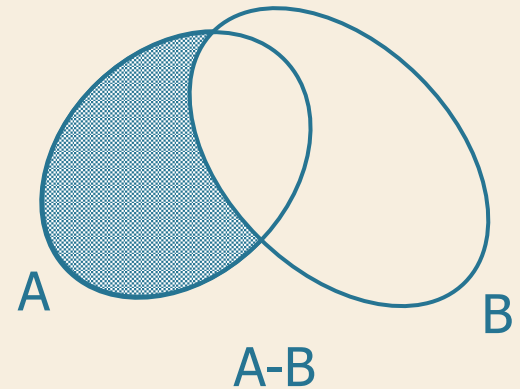
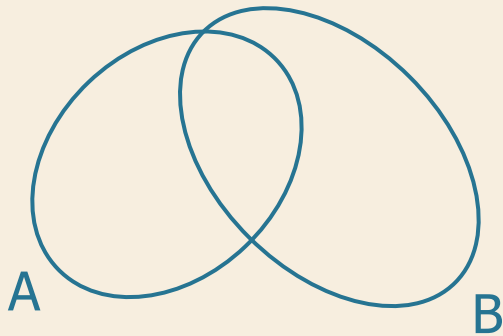


R

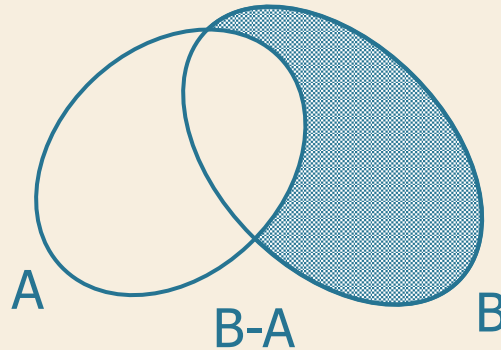
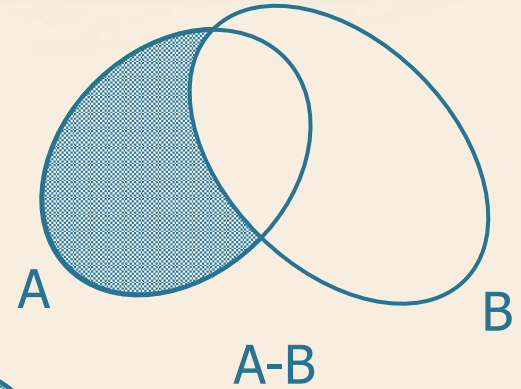
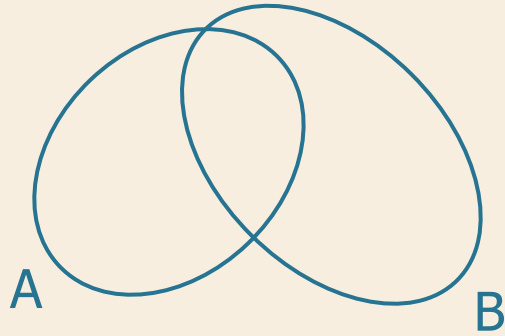
MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica

Differenza

➤ La differenza di due relazioni A e B seleziona tutte le tuple presenti *esclusivamente* in A



Differenza



$$A-B \neq B-A$$

Differenza: esempio (n.1)

➤ *Trovare i docenti di corsi di laurea ma non di master*

Differenza: esempio (n.1)

DocentiLaurea

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
<i>D104</i>	<i>Bianchi</i>	<i>Elettronica</i>

DocentiMaster

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D101	Rossi	Elettrica

R

MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica
D104	Bianchi	Elettronica

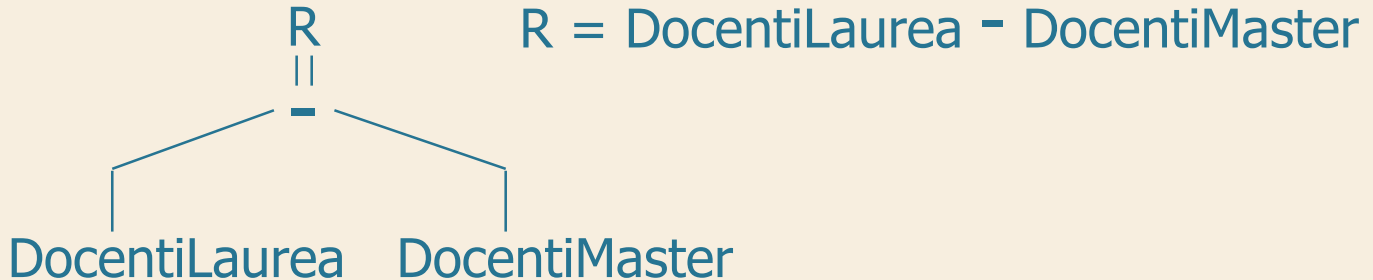
Differenza: definizione

$$R = A - B$$

- La differenza di due relazioni A e B genera una relazione R
 - avente lo stesso schema di A e di B
 - contenente tutte le tuple appartenenti ad A che non appartengono a B
- *Compatibilità*
 - le relazioni A e B devono avere lo stesso schema (numero e tipo degli attributi)
- La differenza *non gode* né della proprietà commutativa, né della proprietà associativa

Differenza: esempio (n.1)

➤ *Trovare i docenti di corsi di laurea ma non di master*



R

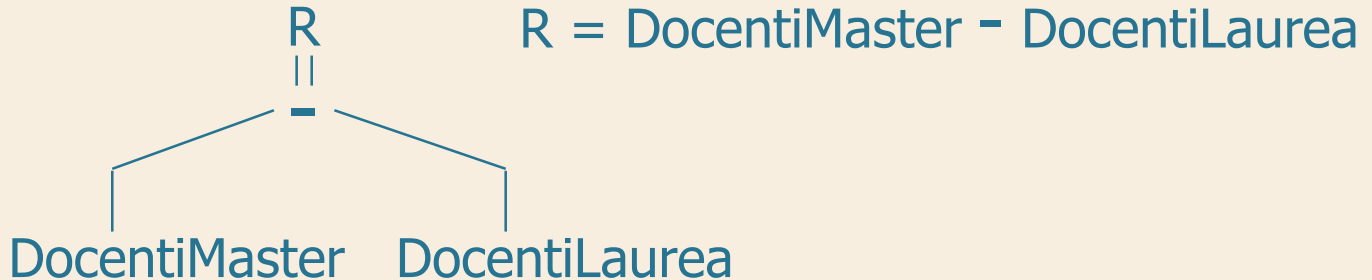
MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica
D104	Bianchi	Elettronica

Differenza: esempio (n. 2)

➤ *Trovare i docenti di corsi di master ma non di laurea*

Differenza: esempio (n. 2)

➤ *Trovare i docenti di corsi di master ma non di laurea*



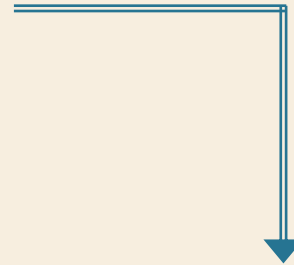
Differenza: esempio (n. 2)

DocentiMaster

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
<i>D101</i>	<i>Rossi</i>	<i>Elettrica</i>

DocentiLaurea

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica



R

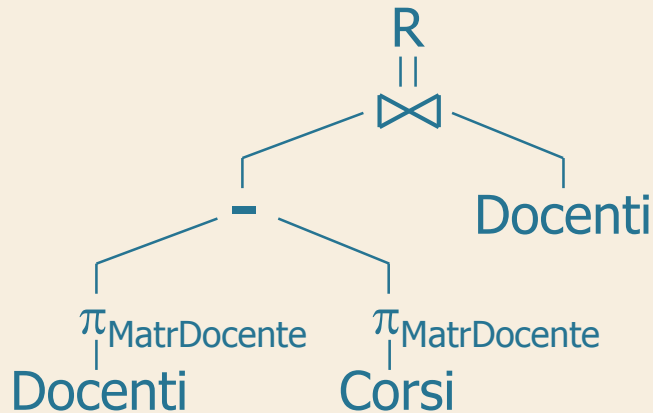
<u>MatrDocente</u>	NomeDoc	Dipartimento
D101	Rossi	Elettrica

Differenza: esempio (n. 3)

➤ *Trovare Matricola, Nome e Dipartimento dei docenti che non tengono corsi*

Differenza: esempio (n. 3)

➤ *Trovare Matricola, Nome e Dipartimento dei docenti che non tengono corsi*



$$R = Docenti \bowtie ((\pi_{MatrDocente} Docenti) - (\pi_{MatrDocente} Corsi))$$

Differenza: esempio (n. 3)

Matricole dei docenti

Docenti

<u>MatrDocente</u>	NomeDoc	Dipartimento
<i>D102</i>	Verdi	Informatica
<i>D105</i>	Neri	Informatica
<i>D104</i>	Bianchi	Elettronica

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	<i>D102</i>
M4880	Sistemi digitali	2	<i>D104</i>
F1401	Elettronica	1	<i>D104</i>
F0410	Basi di dati	2	<i>D102</i>

Matricole dei docenti che tengono almeno un corso

Differenza: esempio (n. 3)

MatrDocente
D102
<i>D105</i>
D104

Differenza →

MatrDocente
D105

MatrDocente
D102
D104

Differenza: esempio (n. 3)

MatrDocente

D105

Docenti

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
D104	Bianchi	Elettronica

Natural Join

R

MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica

- L'anti-join tra due relazioni A e B seleziona tutte le tuple di A *"semanticamente non legate"* a tuple di B
- le informazioni di B non compaiono nel risultato

Anti-join: esempio

➤ *Trovare Matricola, Nome e Dipartimento dei docenti che non tengono corsi*

Anti-join: esempio

Docenti

<u>MatrDocente</u>	NomeDoc	Dipartimento
D102	Verdi	Informatica
<i>D105</i>	<i>Neri</i>	<i>Informatica</i>
D104	Bianchi	Elettronica

Corsi

<u>Codice</u>	NomeCorso	Semestre	MatrDocente
M2170	Informatica 1	1	D102
M4880	Sistemi digitali	2	D104
F1401	Elettronica	1	D104
F0410	Basi di dati	2	D102

R

MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica

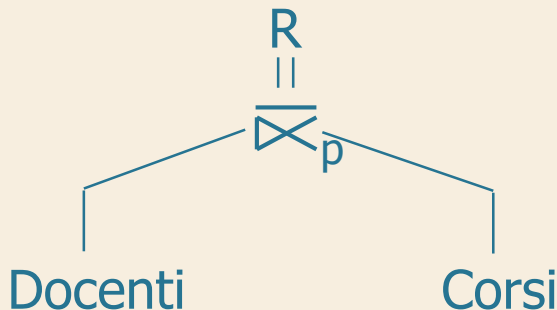
Anti-join: definizione

$$R = A \bar{\bowtie}_p B$$

- L'anti-join di due relazioni A e B genera una relazione R
 - avente lo stesso schema di A
 - contenente tutte le tuple di A per cui non esiste nessuna tupla in B per cui è vero il predicato p
- Il predicato p è espresso nella stessa forma del theta-join e del semi-join
- L'anti-join *non gode* né della proprietà commutativa, né della proprietà associativa

Anti-join: esempio

➤ *Trovare Matricola, Nome e Dipartimento dei docenti che non tengono corsi*



$$R = Docenti \bar{\bowtie}_p Corsi$$

p : $Docenti.MatrDocente = Corsi.MatrDocente$

R

MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica

Divisione: esempio

➤ *Trovare gli studenti che hanno superato l'esame di **tutti** i corsi del primo anno*

EsamiSuperati

<u>MatrStudente</u>	<u>CodCorso</u>
S1	C1
S1	C2
S1	C3
S1	C4
S1	C5
S1	C6
S2	C1
S2	C2
S3	C2
S4	C2
S4	C4
S4	C5

CorsiPrimoAnno

<u>CodCorso</u>
...
...
...
...

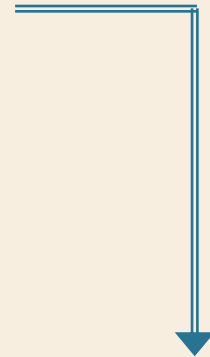
Divisione: esempio (n. 1)

EsamiSuperati

<u>MatrStudente</u>	<u>CodCorso</u>
<i>S1</i>	<i>C1</i>
S1	C2
S1	C3
S1	C4
S1	C5
S1	C6
<i>S2</i>	<i>C1</i>
S2	C2
S3	C2
S4	C2
S4	C4
S4	C5

CorsiPrimoAnno

<u>CodCorso</u>
C1



R

<u>MatrStudente</u>
S1
S2

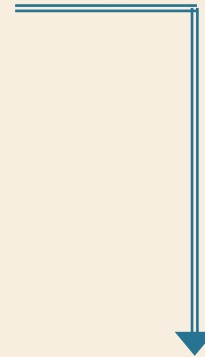
Divisione: esempio (n. 2)

EsamiSuperati

<u>MatrStudente</u>	<u>CodCorso</u>
S1	C1
<i>S1</i>	<i>C2</i>
S1	C3
<i>S1</i>	<i>C4</i>
S1	C5
S1	C6
S2	C1
S2	C2
S3	C2
<i>S4</i>	<i>C2</i>
<i>S4</i>	<i>C4</i>
S4	C5

CorsiPrimoAnno

<u>CodCorso</u>
C2
C4



R

<u>MatrStudente</u>
S1
S4

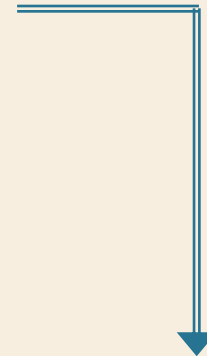
Divisione: esempio (n. 3)

EsamiSuperati

<u>MatrStudente</u>	<u>CodCorso</u>
<i>S1</i>	<i>C1</i>
<i>S1</i>	<i>C2</i>
<i>S1</i>	<i>C3</i>
<i>S1</i>	<i>C4</i>
<i>S1</i>	<i>C5</i>
<i>S1</i>	<i>C6</i>
S2	C1
S2	C2
S3	C2
S4	C2
S4	C4
S4	C5

CorsiPrimoAnno

<u>CodCorso</u>
C1
C2
C3
C4
C5
C6



R

<u>MatrStudente</u>
S1

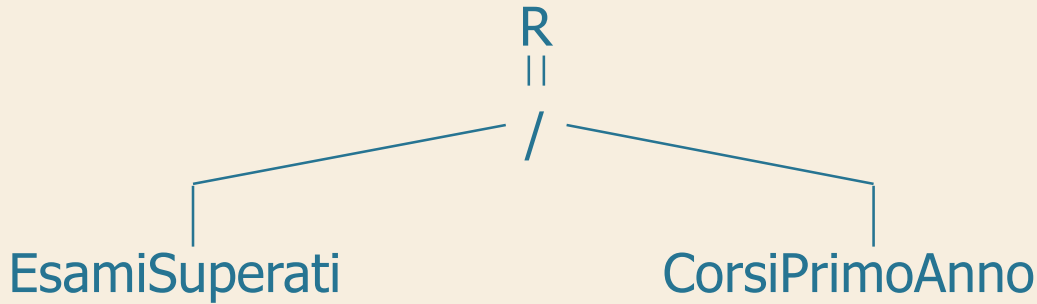
Divisione: definizione

$$R = A / B$$

- La divisione della relazione A per la relazione B genera una relazione R
 - avente come schema *schema(A) - schema(B)*
 - contenente tutte le tuple di A tali che per ogni tupla (Y:y) presente in B esiste una tupla (X:x, Y:y) in A
- La divisione *non gode* né della proprietà commutativa, né della proprietà associativa

Divisione: esempio

➤ *Trovare gli studenti che hanno superato l'esame di **tutti** i corsi del primo anno*



$$R = EsamiSuperati / CorsiPrimoAnno$$

- Sono stati proposti numerosi altri operatori per estendere il potere espressivo dell'algebra relazionale
- estensione con un nuovo attributo, definito da un'espressione scalare
 - $PESO_LORDO = PESO_NETTO + TARA$
 - calcolo di funzioni aggregate
 - max, min, avg, count, sum
 - eventualmente con la definizione di sottoinsiemi in cui raggruppare i dati (GROUP BY di SQL)