



Modello relazionale Operazioni

Prof. Viglietti Francesco
[Www.in4matika.altervista.org](http://www.in4matika.altervista.org)



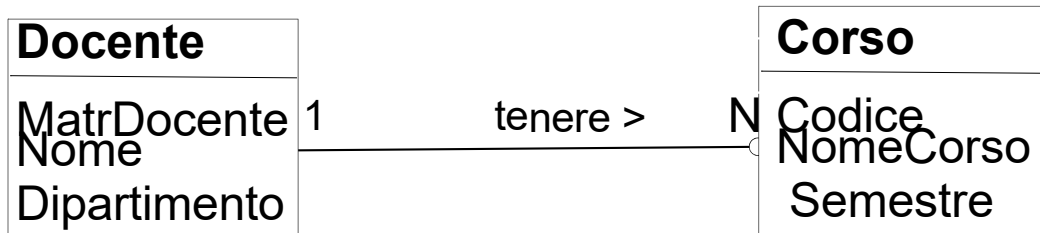
Operazioni Relazionali

Le operazioni relazionali consentono di interrogare il DB, agiscono su una o più relazioni per ottenerne una nuova. In generale le principali sono le seguenti:

- **La selezione:** genera una nuova relazione costituita solo dalle tuple della relazione di partenza che soddisfano una determinata condizione.
- **La proiezione:** genera una nuova relazione estraendo dalla relazione di partenza solo determinate colonne.
- **La congiunzione:** tra due tabelle combina due relazioni con attributi in comune, generando una nuova relazione le cui righe contengono gli attributi comuni delle relazioni di partenza e gli altri attributi non comuni

Per le operazioni relazionali seguenti ci baseremo sul seguente modello:

Operazioni Relazionali



La struttura delle relazioni sarà:

Docenti (MatrDocente {PK}, NomeDoc, Dipartimento)

Corsi (Codice {PK}, NomeCorso, Semestre, MatrDocente {FK})

<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

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

Selezione

Consente di effettuare un “taglio orizzontale” su una relazione, cioè di selezionare solo le righe che soddisfano una data condizione. La relazione ottenuta presenta Grado uguale, cardinalità minore o uguale a quella di partenza

Si denota con il simbolo $R = \sigma_p T$

La selezione genera una relazione R

- avente lo stesso schema di T (tabella di partenza)
- contenente tutte le tuple della relazione T 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

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

Esempio: Trovare i corsi che si tengono nel secondo semestre

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

Proiezione

Consente di effettuare un “taglio verticale” su una relazione, cioè di selezionare solo alcune colonne (attributi). La relazione ottenuta presenta un Grado minore o uguale, ed una Cardinalità minore o uguale a quella di partenza

Si denota con il simbolo $R = \Pi_L T$.

- La proiezione genera una relazione R
 - avente come schema la lista di attributi L (sottoinsieme dello schema di T)
 - contenente tutte le tuple presenti in T
- Esempio: Trovare il nome dei docenti

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

NomeDoc
Verdi
Neri
Bianchi

Esempio 2: Trovare i nomi dei dipartimenti in cui è presente almeno un docente

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

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica



Dipartimento
Informatica
Elettronica

Prodotto cartesiano

Il prodotto cartesiano di due relazioni A e B genera una relazione R, e si indica $R = A \times B$:

–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.

Esempio:Trovare il prodotto cartesiano tra Corsi e Docenti

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

Congiunzione (Join)

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”. Il join è un operatore derivato e può essere espresso utilizzando gli operatori \times , σ , Π

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

Esempio: Trovare le informazioni sui corsi e sui docenti che li tengono

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

Natural Join

Il natural join di due relazioni A e B genera una relazione $R = A \bowtie B$

– 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

Esempio: Trovare le informazioni sui corsi e sui docenti che li tengono

N.B. l'attributo comune MatrDocente è presente una volta sola nello schema della relazione Risultante.

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

Theta Join - Equi 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”. Il theta-join di due relazioni A e B genera una relazione $R = A \bowtie_p B$

–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

Equi-join → caso particolare del theta-join in cui θ è l’operatore di uguaglianza (=) $R = A \bowtie_{=} B$

Esempio: Trovare la matricola dei docenti che sono titolari di almeno due corsi

Theta Join - Equi Join

Il primo passo consiste nell'individuare le operazioni relazionali che ci permettano di soddisfare la richiesta. Viene detto trovare la matricola → dev'esserci per forza una proiezione. Poi siano titolari di almeno due corsi → questa condizione è apparentemente difficile da verificare... Bisogna usare un "trucchetto", duplichiamo i corsi e vediamo di verificare il predicato p.

$$R = \Pi_{C1.MatrDocente}((Corsi\ C1) \bowtie p (Corsi\ C2))$$

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

Corsi C1

Codice	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

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

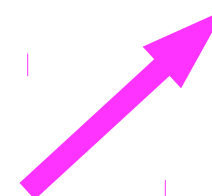
Theta Join 2

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



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

Corsi C1. MatrDocente
D102
D104



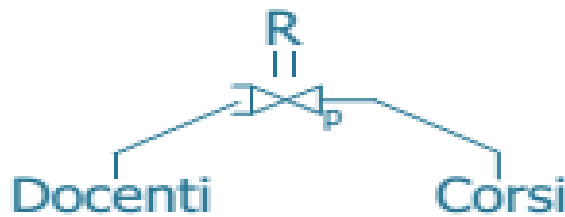
Altre operazioni

Vi sono delle altre operazioni non meno importanti per quanto riguarda il modello relazionale, chiamate in generale **Outer join** esse possono restituire le righe dell'una o dell'altra tabella anche se non vi sono valori comuni. Tali operazioni sono: **Left join** elenca tutte le righe della prima tabella con quelle della seconda che hanno valori uguali agli attributi comuni. **Right join** elenca tutte le righe della seconda tabella con quelle della prima che hanno valori uguali agli attributi comuni. **Full join** combina le caratteristiche del Left join e del Right join. **Self join** si combinano righe della stessa tabella con valori corrispondenti degli attributi. Vediamo ora in dettaglio le suddette operazioni.

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

Esempio: **Trovare le informazioni su tutti i docenti e sui corsi che tengono**



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

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

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
D105	Neri	Informatica	null	null	null	null

Right (outer) Join

- Il right 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

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

– 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 Il full outer-join è commutativo

Riepilogo ... ed operazioni tra insiemi

Quindi le operazioni di Selezione e Proiezione agiscono su una sola tabella, mentre la Congiunzione agisce su più tabelle.

Queste operazioni possono essere combinate tra loro in base alle richieste.

Se le tabelle hanno una struttura tabellare omogenea, si applicano le operazioni tra gli insiemi: Unione, Intersezione e Differenza.

Unione

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

DocentiLaurea

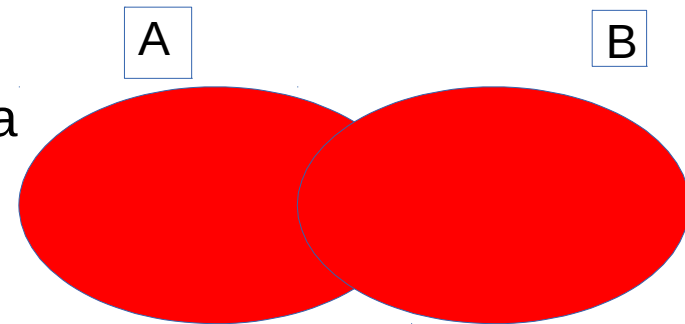
MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica

DocentiMaster

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D101	Rossi	Elettrica



Francesco Viglietti



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. Si indica $R = A \cap B$.

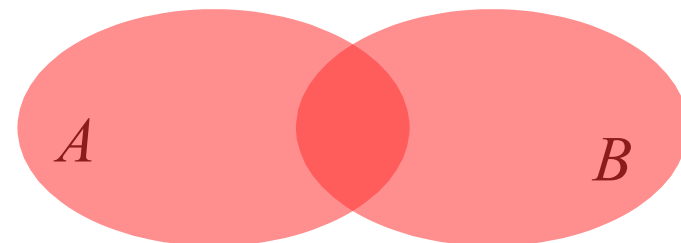
L'intersezione di due relazioni A e B genera una relazione R

- 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



Esempio: Trovare le informazioni relative ai docenti sia di corsi di laurea, sia di master

DocentiLaurea

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D105	Neri	Informatica
D104	Bianchi	Elettronica



MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica

DocentiMaster

MatrDocente	NomeDoc	Dipartimento
D102	Verdi	Informatica
D101	Rossi	Elettrica

Differenza

$R = A - B$. 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

Esempio 1: Trovare i docenti di corsi di laurea ma non di master

MatrDocente	NomeDoc	Dipartimento
D105	Neri	Informatica
D104	Bianchi	Elettronica

Esempio 2: Trovare i docenti di corsi di master ma non di laurea

MatrDocente	NomeDoc	Dipartimento
D101	Rossi	Elettrica

Esempio 1

Query : Nomi dei clienti che hanno ordinato più di 100 pezzi per almeno un articolo

Nome	C#	Città
Rossi	C1	Roma
Rossi	C2	Roma
Bianchi	C3	Milano

Ordini

C#	A#	Npezzi
C1	A1	100
C2	A2	200
C3	A2	150
C1	A3	100
C1	A2	200

Esempio 1

Clienti \triangleright \triangleleft Ordini

Nome	C#	Città	A#	Npezzi
Rossi	C1	Roma	A1	100
Rossi	C1	Roma	A2	200
Rossi	C1	Roma	A3	100
Rossi	C2	Milano	A2	200
Bianchi	C3	Roma	A2	150

$\sigma_{N\text{-pezzi} > 100}(\text{Clienti } \triangleright \triangleleft \text{Ordini})$

Nome	C#	Città	A#	Npezzi
Rossi	C1	Roma	A2	200
Rossi	C2	Milano	A2	200
Bianchi	C3	Roma	A2	150

$\pi \text{ Nome}(\sigma_{N\text{pezzi} > 100}(\text{Clienti } \triangleright \triangleleft \text{Ordini}))$

Nome
Rossi
Bianchi

Esempio 2

Query : Nomi e città dei clienti che hanno ordinato almeno 100 pezzi per almeno un articolo con prezzo superiore a 2 euro. Utilizza le tabelle clienti e ordini dell'esercizio precedente.

Articoli

A#	Denom.	Prezzo
A1	Piatto	3
A2	Bicchiere	2
A3	tazza	4

π Nome, Città($\sigma_{N-pezzi \geq 100 \wedge Prezzo > 2}((Clienti \bowtie Ordini) \bowtie Articoli)$)

Esempio 2

(Clienti \triangleright \triangleleft Ordini) \triangleright \triangleleft Articoli

Nome	C#	Città	A#	N-pezzi	Denom.	Prezzo
Rossi	C1	Roma	A1	100	Piatto	3
Rossi	C1	Roma	A2	200	Bicchiere	2
Rossi	C1	Roma	A3	100	Tazza	4
Rossi	C2	Milano	A2	200	Bicchiere	2
Bianchi	C3	Roma	A2	150	Bicchiere	2

Nome	C#	Città	A#	N-pezzi	Denom.	Prezzo
Rossi	C1	Roma	A1	100	Piatto	3
Rossi	C1	Roma	A3	100	Tazza	4

π Nome, Città (σ N-pezzi \geq 100 \wedge Prezzo $>$ 2 ((Clienti \triangleright \triangleleft Ordini) \triangleright \triangleleft Articoli))

Nome	Città
Rossi	Roma

Esercizio

Le informazioni relative alle attività sportive studentesche devono essere organizzate in una base di dati. Gli studenti, dei quali si conservano le informazioni anagrafiche, frequentano gli Istituti superiori, e possono partecipare a una o più manifestazioni sportive. (specialità sportive diverse, giornate diverse, campionati che durano per mesi o gare di un giorno.

Per ogni attività sportiva le scuole indicano un professore che svolga la funzione di riferimento e di allenatore: ogni professore segue una sola manifestazione ma una stessa manifestazione può essere seguita da professori diversi di scuole diverse.

1 Definire il modello SQL del database attraverso il modello E/R

Esercizio

2 implementare le seguenti interrogazioni:

- Numero degli studenti che partecipano a una determinata manifestazione sportiva
- Elenco anagrafico degli allenatori di una attività sportiva
- Elenco delle scuole (solo il nome) con il numero di studenti che partecipano alle attività sportive
- Elenco delle scuole (con nome, indirizzo e telefono) con studenti che partecipano a una determinata manifestazione sportiva
- Elenco degli allenatori (cognome e nome) e scuole (denominazione) di appartenenza in ordine alfabetico
- Numero degli studenti partecipanti di una determinata scuola per ciascuna manifestazione sportiva