

Basi di Dati

Il linguaggio SQL (terz parte)

**Marco Maratea
Laurea in Informatica, DeMaCS, UNICAL**

28 Novembre 2025



Interrogazioni nidificate

- le condizioni atomiche permettono anche
 - il confronto fra un attributo (o più, vedremo poi) e il risultato di una sottointerrogazione
 - quantificazioni esistenziali



- nome e reddito del padre di Franco

```
select Nome, Reddito  
from Persone, Paternita  
where Nome = Padre and Figlio = 'Franco'
```

```
select Nome, Reddito  
from Persone  
where Nome = ( select Padre  
                from Paternita  
                where Figlio = 'Franco' )
```



Interrogazioni nidificate, commenti

- La forma nidificata è “meno dichiarativa”, ma talvolta più leggibile (richiede meno variabili)
- La forma ‘piana’ e quella nidificata possono essere combinate
- Vi sono limitazioni, ad esempio le sottointerrogazioni non possono contenere operatori insiemistici (“l’unione si fa solo al livello esterno”)



- Nome e reddito dei padri di persone che guadagnano più di 20

```
select distinct P.Nome, P.Reddito
from Persone P, Paternita, Persone F
where P.Nome = Padre and Figlio = F.Nome
      and F.Reddito > 20
```

```
select Nome, Reddito
from Persone
where Nome in (select Padre
                 from Paternita
                 where Figlio = any (select Nome
                                      from Persone
                                      where Reddito > 20))
```



- Nome e reddito dei padri di persone che guadagnano più di 20

```
select distinct P.Nome, P.Reddito  
from Persone P, Paternita, Persone F  
where P.Nome = Padre and Figlio = F.Nome  
      and F.Reddito > 20
```

(con forma ‘piana’ e nidificata combinate)

```
select Nome, Reddito  
from Persone  
where Nome in (select Padre  
                  from Paternita, Persone  
                  where Figlio = Nome  
                  and Reddito > 20)
```



Interrogazioni nidificate, commenti, 2

- La prima versione di SQL prevedeva solo la forma nidificata (o strutturata), con una sola relazione in ogni clausola FROM. Insoddisfacente:
 - la dichiaratività è limitata
 - non si possono includere nella target list attributi di relazioni nei blocchi interni



- Nome e reddito dei padri di persone che guadagnano più di 20, con indicazione del reddito del figlio

```
select distinct P.Nome, P.Reddito, F.Reddito
from Persone P, Paternita, Persone F
where P.Nome = Padre and Figlio = F.Nome
      and F.Reddito > 20
```

```
select Nome, Reddito, ****?
from Persone
where Nome in (select Padre
                  from Paternita
                  where Figlio = any (select Nome
                                      from Persone
                                      where Reddito > 20))
```



Paolo Atzeni
Stefano Ceri
Piero Fraternali
Stefano Paraboschi
Riccardo Torlone

Basi di dati

VI edizione

McGraw-Hill Connect

McGraw-Hill

Interrogazioni nidificate, commenti, 3

- regole di visibilità:
 - non è possibile fare riferimenti a variabili definite in blocchi più interni
 - in un blocco si può fare riferimento a variabili definite in blocchi più esterni; la semantica base (prodotto cartesiano, selezione, proiezione) può non funzionare più, vedremo presto



Quantificazione esistenziale

- Ulteriore tipo di condizione
 - **EXISTS (Sottoespressione)**
 - Vera se la sottoespressione non ritorna insieme vuoto



- Le persone che hanno almeno un figlio

```
select *  
from Persone  
where exists (  
    select *  
    from Paternita  
    where Padre = Nome) or  
exists (  
    select *  
    from Maternita  
    where Madre = Nome)
```



- I padri i cui figli guadagnano tutti più di 20

```
select distinct Padre
from Paternita Z
where not exists (
    select *
    from Paternita W, Persone
    where W.Padre = Z.Padre
        and W.Figlio = Nome
        and Reddito <= 20)
```



- I padri i cui figli guadagnano tutti più di 20

```
select distinct Padre
from Paternita
where not exists (
    select *
    from Persone
    where Figlio = Nome
    and Reddito <= 20)
```

NO!!!



Semantica delle espressioni “correlate”

- L'interrogazione interna viene eseguita una volta per ciascuna ennupla dell'interrogazione esterna



Visibilità

- scorretta:

```
select *
from Impiegato
where Dipart in (select Nome
                  from Dipartimento D1
                  where Nome = 'Produzione') or
          Dipart in (select Nome
                  from Dipartimento D2
                  where D2.Citta = D1.Citta)
```



Disgiunzione e unione (ma non sempre)

- Persone che hanno reddito > 30, e figli di un padre che guadagna > 30

```
select * from Persone where Reddito > 30
```

```
union
```

```
select F.*
```

```
from Persone F, Paternita, Persone P
```

```
where F.Nome = Figlio and Padre = P.Nome and P.Reddito > 30
```

```
select *
```

```
from Persone F
```

```
where Reddito > 30 or
```

```
exists (select *
```

```
from Paternita, Persone P
```

```
where F.Nome = Figlio and Padre = P.Nome and P.Reddito > 30)
```



Differenza e nidificazione

- Trovare i Nomi degli Impiegati che non compaiono anche come Cognomi

```
select Nome from Impiegato  
except  
select Cognome as Nome from Impiegato
```

```
select Nome  
from Impiegato I  
where not exists (select *  
                  from Impiegato  
                  where Cognome = I.Nome)
```

Versioni recenti di SQL sono meno limitate sugli operatori insiemistici: non serve stesso nome degli attributi, ma basta numero, ordine e domini compatibili.



Operatori aggregati

- Nelle espressioni della target list possiamo avere anche espressioni che calcolano valori a partire da insiemi di ennuple:
 - conteggio, minimo, massimo, media, totale
 - sintassi base (semplificata):
 $\text{Funzione ([DISTINCT] *)}$
 $\text{Funzione ([DISTINCT] Attributo)}$



Operatori aggregati: COUNT

- Il numero di figli di Franco

```
select count(*) as NumFigliDiFranco  
from Paternita  
where Padre = 'Franco'
```

- l'operatore aggregato (`count`) viene applicato al risultato dell'interrogazione:

```
select *  
from Paternita  
where Padre = 'Franco'
```



Paternità	Padre	Figlio
	Sergio	Franco
	Luigi	Olga
	Luigi	Filippo
	Franco	Andrea
	Franco	Aldo

NumFigliDiFranco
2



COUNT DISTINCT

select count(*) from persone → 4

select count(distinct Reddito) from persone → 2

Persone	Nome	Età	Reddito
	Andrea	27	21
	Aldo	25	35
	Maria	55	21
	Anna	50	35



Altri operatori aggregati

- SUM, AVG, MAX, MIN
- Media dei redditi dei figli di Franco

```
select avg(redotto)
from persone join paternita on nome=figlio
where padre='Franco'
```



COUNT e valori nulli

select count(*) from persone → 4

select count(redotto) from persone --> 3

select count(distinct redotto) from persone → 2

Persone	Nome	Età	Reddito
	Andrea	27	21
	Aldo	25	NULL
	Maria	55	21
	Anna	50	35



Operatori aggregati e valori nulli

select avg(redotto) as redditomedio from persone → 34

Persone	Nome	Età	Reddito
	Andrea	27	30
	Aldo	25	NULL
	Maria	55	36
	Anna	50	36



Operatori aggregati e target list

- un'interrogazione scorretta:

```
select nome, max(redito)  
from persone
```

- di chi sarebbe il nome? La target list deve essere omogenea, mentre nella query precedente abbiamo un attributo ed un valore

```
select min(eta), avg(redito)  
from persone
```



Massimo e nidificazione

- La persona (o le persone) con il reddito massimo

```
select *  
from persone  
where reddito = (      select max(reddito)  
                           from persone)
```



Operatori aggregati e raggruppamenti

- Le funzioni possono essere applicate a partizioni delle relazioni
- Clausola **GROUP BY**:

GROUP BY listaAttributi



Operatori aggregati e raggruppamenti

- Il numero di figli di ciascun padre

```
select Padre, count(*) AS NumFigli
from Paternita
group by Padre
```

Paternita	Padre	Figlio
	Sergio	Franco
	Luigi	Olga
	Luigi	Filippo
	Franco	Andrea
	Franco	Aldo

Padre	NumFigli
Sergio	1
Luigi	2
Franco	2



Semantica di interrogazioni con operatori aggregati e raggruppamenti

1. interrogazione senza `group by` e senza operatori aggregati

`select *`

`from Paternita`

2. si raggruppa e si applica l'operatore aggregato a ciascun gruppo



scorretta

```
select padre, avg(f.reddito), p.reddito
from persone f join paternita on figlio = f.nome join
     persone p on padre =p.nome
group by padre
```

corretta

```
select padre, avg(f.reddito), p.reddito
from persone f join paternita on figlio = f.nome join
     persone p on padre =p.nome
group by padre, p.reddito
```



Condizioni sui gruppi

- I padri i cui figli hanno un reddito medio maggiore di 25; mostrare padre e reddito medio dei figli

```
select padre, avg(f.reddito)
from persone f join paternita on figlio = nome
group by padre
having avg(f.reddito) > 25
```

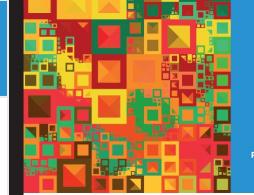
In presenza di group by la target list ‘diventa’ omogenea



WHERE o HAVING?

- I padri i cui figli sotto i 30 anni hanno un reddito medio maggiore di 20

```
select padre, avg(f.reddito)
from persone f join paternita on figlio = nome
where eta < 30
group by padre
having avg(f.reddito) > 20
```



Group by e valori nulli

A	B
1	11
2	11
3	null
4	null

B
11
null

select B, count (*)
from R group by B

A
1
2
3
4

select A, count (*)
from R group by A

select A, count (B)
from R group by A

A
1
2
3
4