

Universita' di Milano Bicocca
Corso di Basi di dati 1 in eLearning

C. Batini

7. SQL DML

7.8 Clausola Having

Condizioni sui gruppi

- Tramite la *Group by* le righe di una tabella sono raggruppate in sottoinsiemi
- Pero', fino ad ora non e' possibile esprimere condizioni su sottoinsiemi
- Vedi esempio →

Condizioni sui gruppi: esempio

- I padri i cui figli hanno un reddito medio maggiore di 25
- Non possiamo, fino ad ora, esprimere condizioni sui gruppi, ma solo operatori aggregati

Padre	Figlio	Reddito
Sergio	Franco	50
Luigi	Olga	30
	Filippo	30
Franco	Andrea	40
	Aldo	40

Condizioni sui gruppi: clausola HAVING

La clausola HAVING permette di esprimere condizioni sui gruppi, applicate a ogni insieme di n-ple risultato della applicazione del GROUP BY.
Sintassi:

```
SELECT target list  
FROM ...  
WHERE ...  
GROUP BY insieme attributi  
HAVING condizione
```

Condizioni sui gruppi: clausola HAVING - esempio

- I padri i cui figli hanno un reddito medio maggiore di 25

```
select Padre, avg (Reddito)
from Persone Join Paternita on Figlio = Nome
group by Padre
having avg(Reddito)>25
```

Mette insieme
Padri e figli

Seleziona i gruppi dei figli che
hanno reddito medio > 25

Raggruppa i figli
per ogni padre

Quando applicare la WHERE e quando la HAVING?

Dato lo schema

PATERNITA' (Padre, Figlio, Eta' figlio, Reddito)

I padri i cui figli sotto i 30 anni hanno un reddito medio maggiore di 25

Domanda 7.8.1: Scrivi la interrogazione in SQL

Quando applicare la WHERE e quando la HAVING?

- I padri i cui figli sotto i 30 anni hanno un reddito medio maggiore di 25

Due condizioni

```
select padre, avg(reddito)
from persone join paternita on figlio = nome
where eta < 30
group by padre
having avg(reddito) > 25
```

Nella clausola WHERE predicato su
operatore elementare

Nella clausola HAVING predicato su
operatore aggregato

Sintassi della select, riassumiamo

SelectSQL ::=

select ListaAttributi o Espressioni

from ListaTabelle

[where CondizioniSemplici]

[group by

ListaAttributiDiRaggruppamento]

[having CondizioniAggregate]

[order by ListaAttributiDiOrdinamento]

Esercizio 7.8.1

Dato lo schema

Cliente (CodFiscaleCliente, CognomeCliente,
Mese, TipoProdotto, Spesa)

che riporta per ogni cliente di una azienda, le
spese effettuate ogni mese di un anno per i vari
tipi di prodotti acquistati (es. CD, penne, ecc.)

Scrivere una interrogazione che riporti i clienti
per prodotto acquistato con la spesa totale, per
il solo mese di Giugno, per i soli prodotti per cui
la spesa totale sia stata superiore a 1.000 €

Interrogazioni di tipo insiemistico

Unione, intersezione e differenza

- La `select` da sola non permette di fare unioni, intersezioni, differenze; serve un costrutto esplicito:

`select ...`

`Union /`

`Intersect /`

`Except`

`[all]`

`select ...`

→ Unione

→ Intersezione

→ Differenza

Semantica degli operatori

E' differente rispetto agli altri operatori:

a. Negli operatori insiemistici i duplicati vengono eliminati (a meno che si usi **all**, vedi in seguito) anche dalle proiezioni!

Questo perche':

1. Sono operatori insiemistici, e devono perciò restituire insiemi
2. L'implementazione e' efficiente

Semantica degli operatori

Al contrario di quanto accade in algebra
relazionale,

gli schemi su cui vengono effettuate le
operazioni possono anche essere diversi:
l'unica condizione e' che gli attributi
devono essere in pari numero e devono
avere domini compatibili

Per l'ordine, vale la posizione degli attributi



Schema di riferimento

Maternita' (Madre, Figlio)

Persone (Nome, Eta', Reddito)

Paternita' (Padre, Figlio)



Notazione posizionale - 1

```
select padre  
from paternita  
union  
select madre  
from maternita
```

Maternita' (Madre, Figlio)
Persone (Nome, Eta', Reddito)
Paternita' (Padre, Figlio)

- quali nomi per gli attributi del risultato?
 - quelli del primo operando → Padre



Notazione posizionale, 2

```
select padre, figlio  
from paternita  
union
```

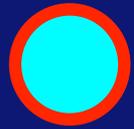
```
select figlio, madre  
from maternita
```

Accoppia padri-figli
Con figli-madri,
scorretta

```
select padre, figlio  
from paternita  
union
```

```
select madre, figlio  
from maternita
```

Corretta



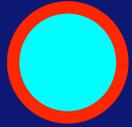
Notazione posizionale, 3

- Anche con le ridenominazioni non cambia niente:

```
select padre as genitore, figlio  
from paternita  
union  
select figlio, madre as genitore  
from maternita
```

- Corretta:

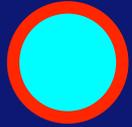
```
select padre as genitore, figlio  
from paternita  
union  
select madre as genitore, figlio  
from maternita
```



Differenza

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

- Trova i nomi di impiegati che non sono anche cognomi di qualche impiegato
- Vedremo che si può anche esprimere con `select nidificate`



Intersezione

```
select Nome  
from Impiegato  
intersect  
select Cognome as Nome  
from Impiegato
```

- Trova i nomi degli impiegati che sono anche cognomi
- Equivale a:

```
select I.Nome  
from Impiegato I, Impiegato J  
where I.Nome = J.Cognome
```

Approfondimento

- Se alcuni aspetti dell'argomento degli operatori insiemistici non ti fossero chiari, studia la sezione 4.3.5 del testo di riferimento.

Esercizio 7.8.2

- Prova ad eseguire per conto tuo le interrogazioni proposte nella sezione 4.3.5 del testo di riferimento e confronta poi la tua soluzione con quella de llibro

Interrogazioni complesse (o nidificate)

Nelle interrogazioni semplici ...

- Le condizioni atomiche permettono sempre e soltanto di confrontare valori elementari (al massimo, di aggregati di gruppi di valori)
- Es. `WHERE Eta' > 30`
- Es. `WHERE Eta' > 25`
`HAVING Avg(Reddito) >= 30`

Estensione alle interrogazioni semplici

- E' utile estendere i confronti a sintassi del tipo

WHERE *Attributi* o *ListadiAttributi*

OperatoreDiConfronto SELECT ..

In cui, come vedremo, possiamo confrontare valori elementari con il risultato di istruzioni SELECT.

Esercizio 7.8.1

- Esegui l'esercizio 4.7 del libro di riferimento, domanda 9.

Concetti introdotti

- Clausola **HAVING**
- Operatore insiemistico
- Operatore **UNION**
- Operatore **INTERSECT**
- Operatore **EXCEPT**
- Interrogazione nidificata (motivazioni)