



Introduzione al linguaggio SQL

Giorgio Bar – giorgio.bar@to.infn.it

Scuola di Specializzazione in Fisica Medica – Università di Torino

Anno accademico 2023/2024 (anno solare 2024/2025)



Istruzione SELECT: operatori aggregati e interrogazioni con raggruppamento

Base dati di esempio

PRODOTTI (codP, NomeP, Colore, Peso, Magazzino)

<u>CodP</u>	NomeP	Colore	Peso	Magazzino
P1	Trion	Rosso	40	Torino
P2	Speed	Giallo	48	Milano
P3	Airtech	Blu	48	Roma

ORDINI (codF, codP, Qta)

<u>CodP</u>	<u>CodF</u>	Qta
P1	F2	300
P2	F2	400
P2	F3	200

FORNITORI (codF, NomeF, Rating, Sede)

<u>CodF</u>	NomeF	Rating	Sede
F1	Atlante	2	Torino
F2	Oceano	1	Milano
F3	Crono	3	Milano

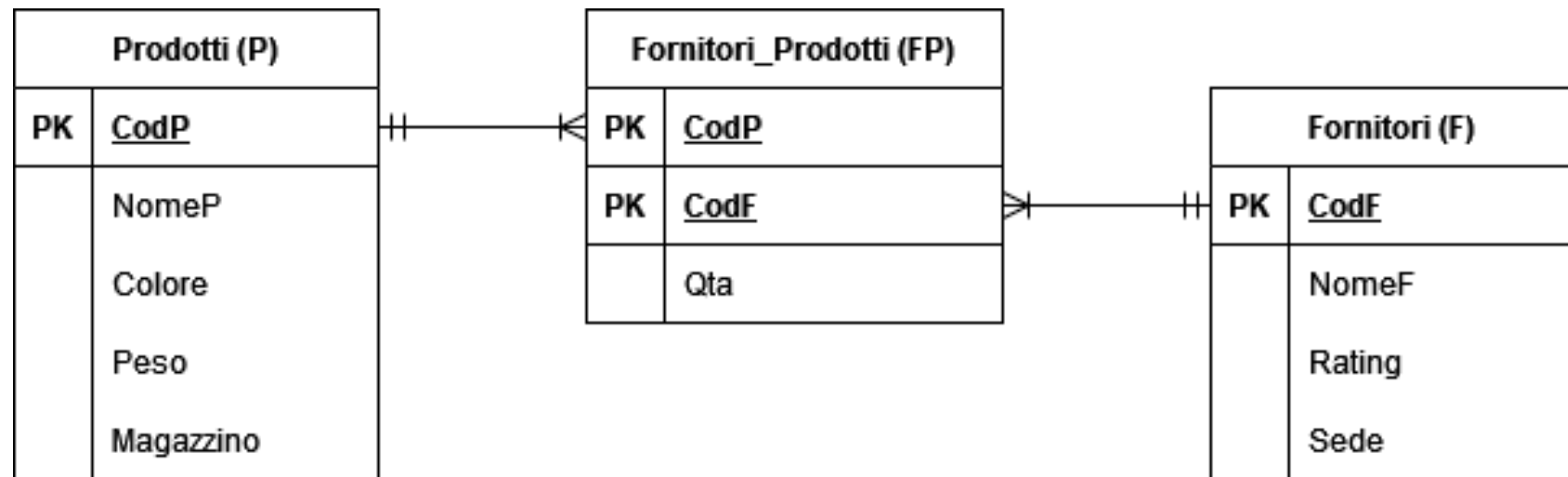
La tabella ORDINI mette in relazione i prodotti con i fornitori che li forniscono

Base dati di esempio

PRODOTTI (codP, NomeP, Colore, Peso, Magazzino)

FORNITORI (codF, NomeF, Rating, Sede)

ORDINI (codF, codP, Qta)



Struttura dell'istruzione SELECT

```
SELECT [DISTINCT] lista_attributi_o_funzioni_da_visualizzare  
FROM elenco_tabelle  
[ WHERE condizioni_di_tupla ]  
[ GROUP BY attributi_di_raggruppamento ]  
[ HAVING condizioni_su_aggregati ]  
[ ORDER BY attributi_di_ordinamento ];
```

Operatori aggregati

Una estensione di SQL rispetto all'algebra relazionale.

- In algebra relazionale le condizioni vengono valutate su ciascuna tupla, indipendentemente da tutte le altre.
- In SQL è possibile valutare proprietà che dipendono da insiemi di tuple, utilizzando gli operatori aggregati.
- Questi vengono gestiti come una estensione delle normali interrogazioni: l'operatore aggregato viene **applicato alla tabella che si ottiene considerando solo le clausole FROM e WHERE della interrogazione.**

Operatori aggregati

- Un operatore aggregato opera su un insieme di valori e produce come risultato un unico valore (aggregato)
- Si indica nella clausola SELECT
 - Non si possono indicare anche attributi non aggregati
 - Possono essere richiesti più operatori aggregati contemporaneamente

```
SELECT elenco_operatori_aggregati_da_visualizzare  
FROM elenco_tabelle_da_utilizzare  
[ WHERE condizioni_di_tupla ];
```

<https://www.postgresql.org/docs/10/tutorial-agg.html>

Funzioni aggregate

- Funzioni aggregate disponibili in SQL-2
 - COUNT: conteggio degli elementi in un attributo
 - SUM: somma dei valori di un attributo
 - AVG: media dei valori di un attributo
 - MAX: massimo valore di un attributo
 - MIN: minimo valore di un attributo
- **Ritornano NULL** se non è stata selezionata alcuna tupla, con la sola eccezione della funzione COUNT, che ritorna zero.
- **Ignorano i valori NULL**, con la sola eccezione di COUNT(*)

Funzione COUNT

- Conteggio del numero di elementi di un insieme
- Numero di righe in cui l'attributo possiede un valore diverso da NULL. L'opzione DISTINCT restituisce il numero di valori distinti di un attributo.

`COUNT([DISTINCT] attributo)`

```
SELECT COUNT(CodF)  
FROM FORNITORI F;
```

- Numero di righe di una tabella

`COUNT(*)`

```
SELECT COUNT(*)  
FROM FORNITORI F;
```

Funzione COUNT

- Trovare il numero di fornitori

```
SELECT COUNT(*)  
FROM FORNITORI F;
```

CodF	NomeF	Rating	Sede
F1	Atlante	2	Torino
F2	Oceano	1	Milano
F3	Crono	3	Milano
F4	Prometeo	2	Torino
F5	Ceo	3	Venezia

```
SELECT COUNT(CodF)  
FROM FORNITORI F;
```

```
SELECT COUNT(DISTINCT CodF)  
FROM FORNITORI F;
```



```
SELECT DISTINCT COUNT(CodF)  
FROM FORNITORI F;
```



COUNT(*)
5

Funzione COUNT

Trovare il numero di fornitori che forniscono almeno un prodotto

CodF	CodP	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400

```
SELECT COUNT(CodF)
FROM ORDINI O;
```



COUNT(CodF)
12

Conta il numero di ordini, non di fornitori
(il numero di righe in cui CodF non è NULL)

```
SELECT COUNT(DISTINCT CodF)
FROM ORDINI O;
```



COUNT(CodF)
4

Operatori aggregati e WHERE

Trovare il numero di fornitori che forniscono il prodotto P2

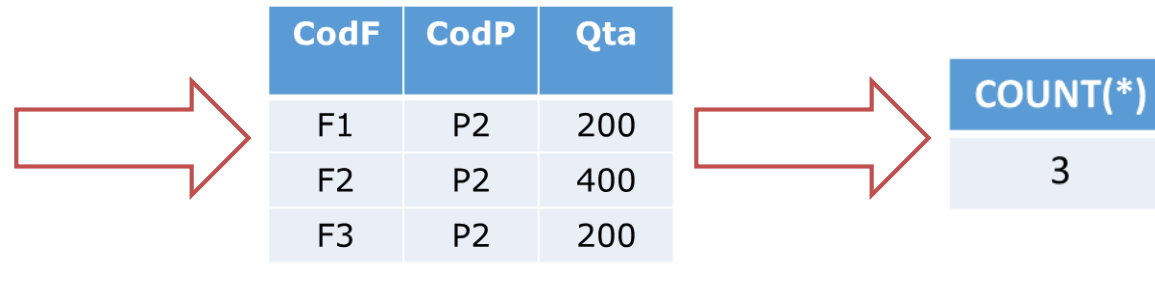
CodF	CodP	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400

Le funzioni aggregate sono valutate solo dopo l'applicazione di tutti i predicati nella WHERE

```
SELECT COUNT(*)  
FROM ORDINI O  
WHERE CodP = 'P2';
```

COUNT(CodF) (?)

COUNT(DISTINCT CodF) (?)



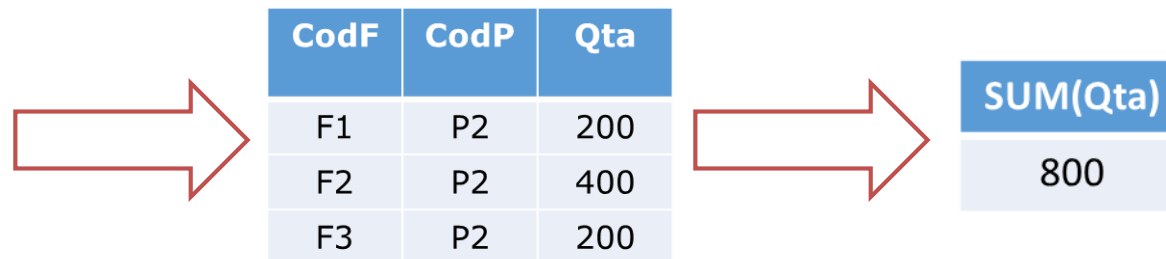
Funzione SUM

- Trovare la quantità totale di pezzi forniti per il prodotto P2

CodF	CodP	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400

Le funzioni aggregate sono valutate solo dopo l'applicazione di tutti i predicati nella WHERE

```
SELECT SUM(Qta)
FROM ORDINI O
WHERE CodP = 'P2';
```



Struttura dell'istruzione SELECT

```
SELECT [DISTINCT] elenco_attributi_da_visualizzare  
FROM elenco_tabelle_da_utilizzare  
[ WHERE condizioni_di_tupla ]  
[ GROUP BY attributi_di_raggruppamento ]  
[ HAVING condizioni_su_aggregati ]  
[ ORDER BY attributi_di_ordinamento ];
```

Raggruppamento

Per ogni prodotto, trovare la quantità totale di pezzi forniti

CodF	CodP	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400



CodF	CodP	Qta
F1	P1	300
F2	P1	300
F1	P2	200
F2	P2	400
F3	P2	200
F1	P3	400
F4	P3	200
F1	P4	200
F4	P4	300
F1	P5	100
F4	P5	400
F1	P6	100




CodP	SUM(Qta)
P1	600
P2	800
P3	600
P4	500
P5	500
P6	100

GROUP BY

Per ogni prodotto, trovare la quantità totale di pezzi forniti

```
SELECT CodP, SUM(Qta)
FROM ORDINI O
GROUP BY CodP;
```



Clausola di raggruppamento
GROUP BY

- Se nella clausola GROUP BY compaiono più attributi, ogni combinazione dei valori di questi attributi costituisce un gruppo.
- L'ordine degli attributi di raggruppamento è influente.
- Nella clausola SELECT possono comparire solo
 - Attributi presenti nella clausola GROUP BY
 - Funzioni aggregate

<https://dev.mysql.com/doc/refman/8.0/en/group-by-handling.html>

GROUP BY e WHERE

- Per ogni prodotto, trovare la quantità totale di pezzi ordinati a fornitori con sede a Milano

```
SELECT ...  
FROM ORDINI O, FORNITORI F  
WHERE O.CodF = F.CodF AND Sede = 'Milano'  
...
```

```
SELECT CodP, SUM(Qta)  
FROM ORDINI O, FORNITORI F  
WHERE O.CodF = F.CodF AND Sede = 'Milano'  
GROUP BY CodP;
```

- I prodotti senza forniture non sono inclusi nel risultato

GROUP BY e SELECT

- Per ogni prodotto, trovare il codice, il **nome** e la quantità totale di pezzi forniti

```
SELECT P.CodP, NomeP, SUM(Qta)
FROM ORDINI O, PRODOTTI P
WHERE O.CodP = P.CodP
GROUP BY P.CodP, NomeP;
```

- Gli attributi univocamente determinati da attributi già presenti nella clausola GROUP BY possono essere aggiunti **senza alterare il risultato** (artificio sintattico)

Struttura dell'istruzione SELECT

```
SELECT [DISTINCT] elenco_attributi_da_visualizzare  
FROM elenco_tabelle_da_utilizzare  
[ WHERE condizioni_di_tupla ]  
[ GROUP BY attributi_di_raggruppamento ]  
[ HAVING condizioni_su_aggregati ]  
[ ORDER BY attributi_di_ordinamento ];
```

Condizione di selezione sui gruppi

- Trovare la quantità totale di pezzi ordinati per i prodotti per cui sono stati ordinati **in totale** almeno 600 pezzi

CodF	CodP	Qta
F1	P1	300
F1	P2	200
F1	P3	400
F1	P4	200
F1	P5	100
F1	P6	100
F2	P1	300
F2	P2	400
F3	P2	200
F4	P3	200
F4	P4	300
F4	P5	400



CodF	CodP	Qta
F1	P1	300
F2	P1	300
F1	P2	200
F2	P2	400
F3	P2	200
F1	P3	400
F4	P3	200
F1	P4	200
F4	P4	300
F1	P5	100
F4	P5	400
F1	P6	100



CodP	SUM(Qta)
P1	600
P2	800
P3	600

La condizione è definita **su valori aggregati**, non è possibile utilizzare la clausola WHERE

Condizione di selezione sui gruppi

- Trovare la quantità totale di pezzi ordinati per i prodotti per cui sono stati ordinati **in totale** almeno 600 pezzi
- La condizione è definita **su valori aggregati**, non è possibile utilizzare la clausola WHERE
- La clausola HAVING permette di specificare condizioni su funzioni aggregate

```
SELECT CodP, SUM(Qta)
FROM ORDINI O
GROUP BY CodP
HAVING SUM(Qta) >= 600;
```

Condizione di selezione sui gruppi

- Trovare il codice dei prodotti rossi forniti da più di un fornitore

O.CodF	O.CodP	O.Qta	P.CodP	P.NomeP	P.Colore	P.Peso	P.Magazzino
F1	P1	300	P1	Trion	Rosso	40	Torino
F2	P1	300	P1	Trion	Rosso	40	Torino
F1	P6	100	P6	Zen	Rosso	42	Torino

```
SELECT O.CodP
FROM ORDINI O, PRODOTTI P
WHERE O.CodP = P.CodP AND Colore = 'Rosso'
GROUP BY O.CodP
HAVING COUNT(*) > 1;
```

O.CodP
P1