



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)



Interrogazioni nidificate

Interrogazioni nidificate

- Una istruzione SELECT contenuta all'interno di un'altra interrogazione
- Permette di suddividere un problema in sottoproblemi più semplici
- Ad esempio, in una clausola WHERE è possibile **confrontare una espressione valutata sulla singola tupla** (nel caso più semplice il nome di un attributo) **con il risultato di una interrogazione**
- Si possono introdurre istruzioni SELECT
 - In un predicato nella clausola WHERE
 - In un predicato nella clausola HAVING
 - Nella clausola FROM (table functions)

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

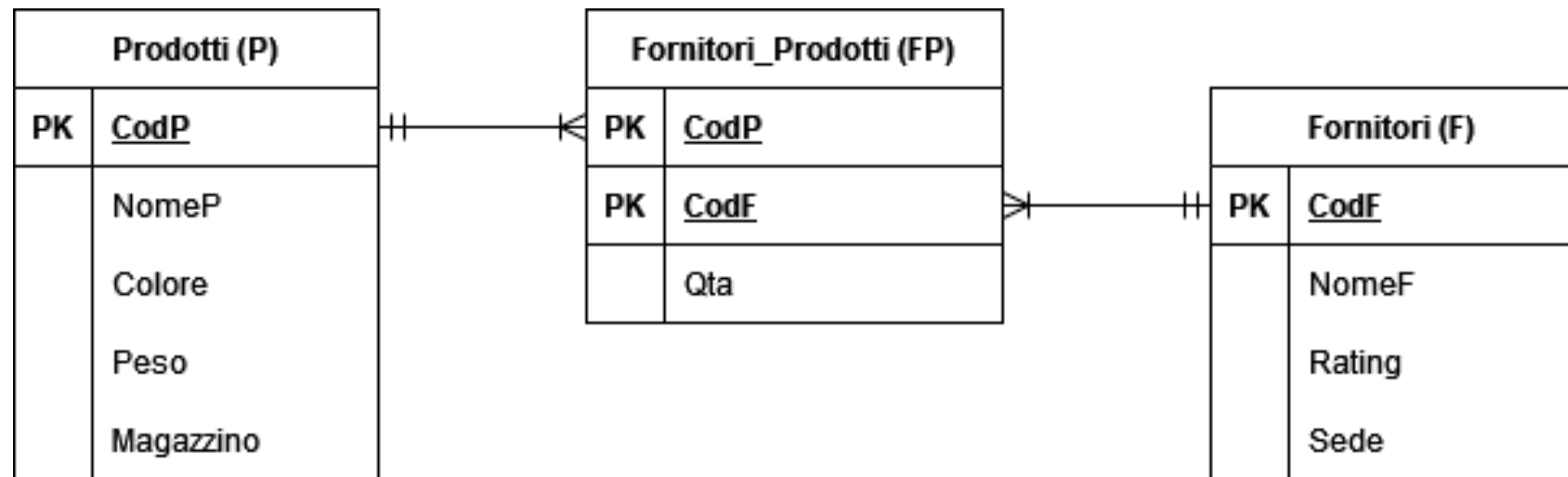
La tabella ORDINI mette in relazione i prodotti con i fornitori che li forniscono

Base dati di esempio

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

FORNITORI (codF, NomeF, Rating, Sede)

ORDINI (codF, codP, Qta)



Operatore IN

- Esprime il concetto di **appartenenza ad un insieme** di valori
attributo IN (interrogazione_nidificata)
- Permette di scrivere l'interrogazione scomponendo il problema in sottoproblemi
- Trovare il nome dei fornitori che forniscono il prodotto P2
 - Trovare i codici dei fornitori del prodotto P2
 - Trovare il nome dei fornitori aventi quei codici

Operatore IN

Trovare il nome dei fornitori che forniscono il prodotto P2

```
SELECT NomeF  
FROM FORNITORI  
WHERE CodF IN
```

Codici dei fornitori del prodotto P2

```
( SELECT CodF  
  FROM ORDINI  
  WHERE CodP = 'P2' );
```

CodF
F1
F2
F3



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

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

Operatore IN

Trovare il nome dei fornitori che forniscono il prodotto P2

```
SELECT NomeF  
FROM FORNITORI
```

Codici dei fornitori del prodotto P2

```
WHERE CodF IN ( SELECT CodF  
                 FROM ORDINI  
                 WHERE CodP = 'P2' );
```

Formulazione equivalente

```
SELECT NomeF  
FROM FORNITORI F, ORDINI O  
WHERE F.CodF = O.CodF  
AND CodP = 'P2';
```

In alcuni casi una interrogazione nidificata può essere espressa mediante un join.

Formulazione equivalente

Trovare il codice dei fornitori il cui rating è **minore del rating massimo**

```
SELECT CodF
FROM FORNITORI
WHERE Rating < ( SELECT MAX(Rating)
                  FROM FORNITORI );
```

Questa interrogazione nidificata
**non può essere espressa
mediante un join.**

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

Operatore IN

Trovare il nome dei fornitori che forniscono almeno un prodotto rosso

3. nomi dei fornitori aventi quei codici fornitore

```
SELECT NomeF  
FROM FORNITORI
```

2. codici dei fornitori di quei prodotti

```
WHERE CodF IN (SELECT CodF  
                FROM ORDINI
```

1. codici dei prodotti rossi

```
                WHERE CodP IN (SELECT CodP  
                                FROM PRODOTTI  
                                WHERE Colore = 'Rosso')));
```

```
SELECT NomeF  
FROM FORNITORI F, ORDINI O, PRODOTTI P  
WHERE O.CodF = F.CodF  
      AND O.CodP = P.CodP  
      AND Colore = 'Rosso';
```

Ma se volessimo trovare il nome dei fornitori che forniscono almeno un prodotto fornito da fornitori di prodotti rossi ...

Concetto di esclusione

- Trovare il nome dei fornitori che **non** forniscono il prodotto P2
- Non è possibile esprimere l'interrogazione mediante il join

```
SELECT NomeF
FROM FORNITORI F, ORDINI O
WHERE F.CodF = O.CodF
AND CodP <> 'P2';
```



- La condizione indicata nella clausola WHERE viene valutata per la singola tupla, se lo stesso fornitore fornisce un altro prodotto oltre a P2 ...
- Questo codice SQL esprime una diversa interrogazione: trovare il nome dei fornitori che forniscono **almeno un prodotto diverso** da P2

Operatore NOT IN


- Esprime il concetto di esclusione da un insieme di valori
`attributo NOT IN (interrogazione_nidificata)`
- Richiede di individuare **l'insieme da escludere**, definito dall'interrogazione nidificata
- Per trovare il nome dei fornitori che **non** forniscono il prodotto P2 occorre escludere dal risultato i fornitori che forniscono il prodotto P2

```
SELECT NomeF  
FROM FORNITORI  
WHERE CodF NOT IN
```

Codici dei fornitori che forniscono P2

```
(SELECT CodF  
FROM ORDINI  
WHERE CodP = 'P2');
```

Operatore NOT IN

- 
- Trovare il nome dei fornitori che forniscono **solo** il prodotto P2
 - Sono fornitori di P2 **che non hanno mai fornito prodotti diversi da P2**
 - Insieme da escludere: fornitori che forniscono **almeno un prodotto diverso** da P2

```
SELECT NomeF  
FROM FORNITORI  
WHERE CodF NOT IN
```

Codici dei fornitori che forniscono
almeno un prodotto diverso da P2

```
(SELECT CodF  
FROM ORDINI  
WHERE CodP <> 'P2' );
```

- Un fornitore che non ha forniture viene selezionato ...

Operatore NOT IN

- Trovare il nome dei fornitori che forniscono **solo** il prodotto P2

```
SELECT NomeF  
FROM FORNITORI F, ORDINI O  
WHERE F.CodF NOT IN
```

Codici dei fornitori che forniscono
almeno un prodotto diverso da P2

```
(SELECT CodF  
FROM ORDINI  
WHERE CodP <> 'P2' )
```

Almeno una fornitura

```
AND F.CodF = O.CodF;
```

- Non serve DISTINCT perché la chiave primaria di ORDINI contiene CodP, se in ORDINI ci sono più tuple per lo stesso fornitore, queste hanno CodP diverso.

Operatore NOT IN

- Trovare il nome dei fornitori che forniscono **solo** il prodotto P2 (alternativa)

```
SELECT NomeF
FROM FORNITORI F
WHERE F.CodF NOT IN (SELECT CodF
                     FROM ORDINI
                     WHERE CodP <> 'P2')
AND F.CodF IN (SELECT CodF
               FROM ORDINI);
```

```
SELECT NomeF
FROM FORNITORI F, ORDINI O
WHERE F.CodF NOT IN (SELECT CodF
                     FROM ORDINI
                     WHERE CodP <> 'P2' )

AND F.CodF = O.CodF;
```

Correlazione tra interrogazioni

- Consente ad una interrogazione nidificata di far riferimento ad attributi definiti in interrogazioni più esterne.
- In particolare, consente ad una interrogazione nidificata di far riferimento al valore di uno o più attributi **della tupla corrente nell'interrogazione esterna.**
- Una condizione di correlazione
 - è indicata nella clausola WHERE dell'interrogazione nidificata
 - è un predicato che lega attributi di tabelle nella FROM dell'interrogazione nidificata con attributi di tabelle nella FROM di interrogazioni **più esterne**
- Un'interrogazione **non** può far riferimento ad attributi
 - Referenziati in un'interrogazione nidificata al suo interno
 - Referenziati in un'interrogazione allo stesso livello

Correlazione tra interrogazioni

- Per ogni prodotto, trovare il codice del fornitore che ne fornisce la quantità massima

```
SELECT CodP, CodF  
FROM ORDINI 01  
WHERE Qta =
```

```
( SELECT MAX(Qta)  
  FROM ORDINI 02  
  WHERE 02.CodP = 01.CodP );
```

Quantità massima per il prodotto **corrente**

Condizione di correlazione

Table function

- Consente di definire una tabella temporanea che può essere utilizzata per ulteriori operazioni di calcolo
- La table function
 - ha la struttura di una SELECT
 - è definita **all'interno di una clausola FROM**
 - può essere referenziata come una normale tabella
 - permette di calcolare più livelli di aggregazione
 - permette di formulare in modo equivalente le interrogazioni che richiedono la correlazione

Aggregati a due livelli

- Dato il seguente schema logico relazionale
STUDENTE (Matricola, AnnoIscrizione)
ESAME (Matricola, CodCorso, Data, Voto)
- Trovare la media massima conseguita da uno studente
- Risoluzione in due passi
 - Trovare la media per ogni studente
 - Trovare il valore massimo della media

```
SELECT MAX(MediaStudente)
FROM ( SELECT Matricola, AVG(Voto) AS MediaStudente
      FROM ESAME
      GROUP BY Matricola ) AS MEDIE ;
```

Table function
(media per ogni studente)