

# Esempio con query annidate e condizioni sui raggruppamenti

## Rappresentanti e fatture ai clienti

Si vogliono organizzare le informazioni relative ai clienti, ai loro rappresentanti e alle fatture emesse. Ogni rappresentante ha tanti clienti, mentre a un cliente corrisponde un solo rappresentante. Per semplicità si supponga di registrare solo la data e l'importo totale di ogni fattura, senza specificare le righe di dettaglio sugli articoli venduti.

Le entità che possono essere individuate nel problema sono:

- *Cliente*, per i clienti ai quali vengono emesse le fatture; attributi di *Cliente* sono: codice del cliente, ragione sociale, indirizzo, telefono, partita IVA.
- *Rappresentante*, per gli agenti ai quali sono assegnati i clienti; suoi attributi sono: codice del rappresentante, cognome, nome, zona assegnata.
- *Fattura*, per le vendite effettuate dai rappresentanti; i suoi attributi sono: numero della fattura, data di emissione, importo.

Tra *Cliente* e *Fattura* esiste un'associazione uno a molti, perché per lo stesso cliente possono essere emesse molte fatture, ma ciascuna fattura è ricevuta da un solo cliente.

Tra *Rappresentante* e *Cliente* esiste un'associazione uno a molti perché ogni rappresentante ha tanti clienti, mentre a un cliente corrisponde un solo rappresentante.

Le associazioni uno a molti si rappresentano introducendo le chiavi esterne nella tabella delle fatture e dei clienti.

Dall'analisi precedente, ricaviamo le tabelle del modello logico del database.

**Rappresentanti** (CodiceRap, CognomeRap, NomeRap, Zona)

**Clienti** (CodiceCli, RagioneSociale, Indirizzo, Telefono, PartitaIVA, *CodiceRap*)

**Fatture** (NumeroFatt, DataFatt, Importo, *CodiceCli*)

## Interrogazioni

*Elenco dei clienti di un rappresentante.*

```
SELECT RagioneSociale, Indirizzo, Telefono
FROM Clienti INNER JOIN Rappresentanti
ON Clienti.CodiceRap = Rappresentanti.CodiceRap
WHERE CognomeRap = [rappresentante da controllare];
```

*Numero dei clienti affidati a un determinato rappresentante.*

```
SELECT COUNT(*) AS NumeroClienti
FROM Clienti INNER JOIN Rappresentanti
ON Clienti.CodiceRap = Rappresentanti.CodiceRap
WHERE CognomeRap = [rappresentante da controllare];
```

*Cognome e nome del rappresentante di un determinato cliente.*

```
SELECT CognomeRap, NomeRap
FROM Clienti INNER JOIN Rappresentanti
ON Clienti.CodiceRap = Rappresentanti.CodiceRap
WHERE RagioneSociale = [cliente da controllare];
```

*Elenco delle fatture di un codice cliente prefissato in ordine di data.*

```
SELECT *
FROM Fatture
WHERE CodiceCli = [codice da controllare]
ORDER BY DataFatt;
```

*Fatturato raggruppato per codice rappresentante.*

```
SELECT CodiceRap, SUM(Importo) AS Fatturato
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
GROUP BY CodiceRap;
```

*Numero delle fatture emesse raggruppate per codice rappresentante.*

```
SELECT CodiceRap, COUNT(*) AS NumeroFatture
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
GROUP BY CodiceRap;
```

*Nome del cliente e data di emissione per la fattura con importo massimo.*

```
SELECT RagioneSociale, DataFatt, Importo
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
WHERE Importo = (SELECT MAX(Importo)
                 FROM Fatture);
```

*Cognome e nome del rappresentante corrispondente alla fattura con importo massimo.*

```
SELECT CognomeRap, NomeRap
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
INNER JOIN Rappresentanti
ON Clienti.CodiceRap = Rappresentanti.CodiceRap
WHERE Importo = (SELECT MAX(Importo)
                 FROM Fatture);
```

*Codice e ragione sociale dei clienti per i quali il fatturato totale è superiore a un valore prefissato.*

```
SELECT CodiceCli, RagioneSociale
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
GROUP BY CodiceCli, RagioneSociale
HAVING SUM(Importo) > [cifra prefissata];
```

*Codice con cognome e nome dei rappresentanti per i quali il numero delle fatture è superiore a un numero prefissato.*

```
SELECT CodiceRap, CognomeRap, NomeRap
FROM Fatture INNER JOIN Clienti
ON Fatture.CodiceCli = Clienti.CodiceCli
INNER JOIN Rappresentanti
ON Clienti.CodiceRap = Rappresentanti.CodiceRap
GROUP BY CodiceRap, CognomeRap, NomeRap
HAVING COUNT(*) > [numero prefissato];
```