

# Basi di Dati

## Esercitazione 2: Interrogazioni in SQL

# DB di riferimento per esempi

---

Consideriamo i seguenti schemi di una base di dati relazionale:

**MODELLI** (cod\_modello, nome, versione, cod\_fabbrica)

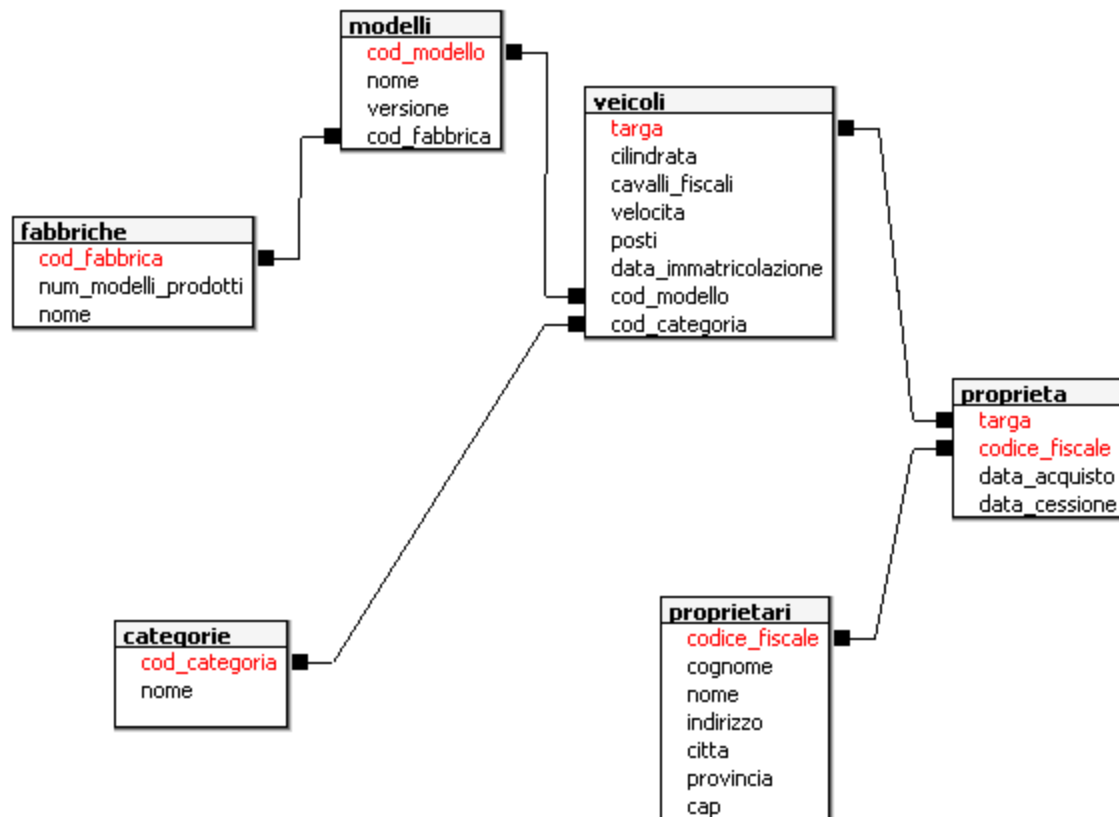
**VEICOLI** (targa, cilindrata, cavalli\_fiscali, velocita, posti, data\_immatricolazione, cod\_modello, cod\_categoria)

**FABBRICHE** (cod\_fabbrica, nome, num\_modelli\_prodotti)

**CATEGORIE** (cod\_categoria, nome)

**PROPRIETARI** (codice\_fiscale, cognome, nome, indirizzo, citta, provincia, cap)

**PROPRIETA** (targa, codice\_fiscale, data\_acquisto, data\_cessione)



## Interrogazioni in SQL

---

Selezionare nome e codice di tutti i Modelli

```
SELECT nome, cod_modello  
FROM Modelli
```

Rinominare le colonne nome, cod\_modello in 'Nome Modello' e 'Codice Modello'

```
SELECT nome AS "Nome Modello", cod_modello AS "Codice Modello"  
FROM Modelli
```

**E' equivalente a:**

```
SELECT nome "Nome Modello", cod_modello "Codice Modello"  
FROM Modelli
```

## Interrogazioni in SQL

---

Selezionare cognome, nome e indirizzo dei proprietari residenti nella provincia di Pisa:

```
SELECT cognome, nome, indirizzo  
FROM Proprietari  
WHERE provincia = 'PI'
```

Fornire le targhe dei veicoli i cui proprietari risiedono nella provincia di Milano:

```
SELECT Proprieta.Targa  
FROM Proprieta  
WHERE Proprieta.codice_fiscale IN (  
    SELECT codice_fiscale  
    FROM Proprietari  
    WHERE provincia = 'MI')
```

## Interrogazioni in SQL

---

Fornire targa, nome e cognome dei proprietari, dei veicoli i cui proprietari risiedono nella provincia di Milano:

```
SELECT Proprieta.Targa, Proprietari.nome, Proprietari.cognome
FROM Proprieta, proprietari
WHERE Proprieta.codice_fiscale = proprietari.codice_fiscale
      AND Proprieta.codice_fiscale IN (
                                SELECT codice_fiscale
                                FROM Proprietari
                                WHERE provincia = 'MI')
```

## Interrogazioni in SQL

---

Partendo dalla query precedente trovare solo i veicoli attualmente in possesso:

```
SELECT Proprieta.Targa, Proprietari.nome, Proprietari.cognome
FROM Proprieta, proprietari
WHERE Proprieta.codice_fiscale = proprietari.codice_fiscale
      AND Proprieta.Data_cessione IS NULL
      AND Proprieta.codice_fiscale IN (
                                SELECT codice_fiscale
                                FROM Proprietari
                                WHERE provincia = 'MI')
```

## Interrogazioni in SQL

---

Selezionare targa e cilindrata dei veicoli la cui categoria e' diversa da 01:

```
SELECT targa, cilindrata  
FROM Veicoli  
WHERE cod_categoria NOT IN ('01')
```

**Oppure**

```
SELECT targa, cilindrata  
FROM Veicoli  
WHERE cod_categoria <> '01'
```



## Interrogazioni in SQL

---

Calcolare il numero dei modelli prodotti da ciascuna fabbrica, ma solo per le fabbriche con almeno 2 modelli:

```
SELECT cod_fabbrica, COUNT(*) "Numero modelli"  
FROM Modelli  
GROUP BY cod_fabbrica  
HAVING COUNT(*) >= 2
```

## Interrogazioni in SQL

---

Visualizzare per ciascun veicolo la targa, il codice e il nome della categoria:

```
SELECT targa, Categorie.cod_categoria, Categorie.nome "Nome Categoria"  
FROM Veicoli, Categorie  
WHERE Veicoli.cod_categoria = Categorie.cod_categoria
```

**Oppure**

```
SELECT targa, Categorie.cod_categoria, Categorie.nome "Nome Categoria"  
FROM Veicoli NATURAL JOIN Categorie
```

## Interrogazioni in SQL

---

Visualizzare il nome delle fabbriche che producono il modello 'PUNTO':

```
SELECT Fabbriche.nome  
FROM Modelli, Fabbriche  
WHERE Modelli.cod_fabbrica = Fabbriche.cod_fabbrica  
AND Modelli.nome = 'PUNTO'
```

**Oppure**

```
SELECT Fabbriche.nome  
FROM Modelli INNER JOIN Fabbriche  
ON Modelli.cod_fabbrica = Fabbriche.cod_fabbrica  
WHERE Modelli.nome = 'PUNTO'
```

## Interrogazioni in SQL

---

Visualizzare per ogni categoria il numero di veicoli:

```
SELECT Categorie.cod_categoria, COUNT(*) AS "Numero veicoli"  
FROM Veicoli, Categorie  
WHERE Veicoli.cod_categoria = Categorie.cod_categoria  
GROUP BY Categorie.cod_categoria
```

**Oppure**

```
SELECT Categorie.cod_categoria, COUNT(*) AS "Numero veicoli"  
FROM Veicoli INNER JOIN Categorie  
ON Veicoli.cod_categoria = Categorie.cod_categoria  
GROUP BY Categorie.cod_categoria
```

**Oppure**

```
SELECT Categorie.cod_categoria, COUNT(*) AS "Numero veicoli"  
FROM Veicoli NATURAL JOIN Categorie  
GROUP BY Categorie.cod_categoria
```



## Interrogazioni in SQL

---

Visualizzare le targhe dei veicoli immatricolati dopo il 1 gennaio 1995

```
SELECT targa  
FROM Veicoli  
WHERE data_immatricolazione > '1995-01-01'
```

# Interrogazioni in SQL

---

Visualizzare le targhe dei veicoli immatricolati nel 2001

```
SELECT targa  
FROM Veicoli  
WHERE EXTRACT (YEAR FROM data_immatricolazione) = 2001
```

**NOTA:**

**EXTRACT (YEAR FROM x):**  
estrae il valore dell'anno da un valore x di tipo DATE

## Interrogazioni in SQL

---

Selezionare i cognomi dei proprietari che sono anche nomi (di qualche proprietario)

```
SELECT Cognome  
FROM Proprietari  
INTERSECT  
SELECT Nome  
FROM Proprietari
```

Selezionare i nomi dei proprietari che non sono anche cognomi (di qualche proprietario)

```
SELECT Nome  
FROM Proprietari  
EXCEPT  
SELECT Cognome  
FROM Proprietari
```

## Interrogazioni in SQL

---

Selezionare i veicoli la cui targa ha un 2 come quarto carattere:

```
SELECT *  
FROM Veicoli  
WHERE targa LIKE '_ _ _2%'
```

Selezionare i veicoli la cui targa inizia o termina con la lettera A:

```
SELECT *  
FROM Veicoli  
WHERE targa LIKE 'A%' OR targa LIKE '%A'
```

\_ rappresenta un confronto con un carattere arbitrario

% rappresenta un confronto con una stringa di lunghezza arbitraria (eventualmente nulla)

**ILIKE** (case-insensitive) presente in Postgresql ma non SQL standard





## JOIN Esplicito:

L'uso del JOIN Esplicito **[INNER JOIN]** permette di distinguere le condizioni che compaiono nell'interrogazione :

- condizioni di JOIN
- condizioni di selezione fra le righe

```
SELECT listaAttributi  
FROM Tabella1 {[INNER] JOIN Tabella2 ON CondDiJoin}  
[WHERE Altracondizione]
```

## Interrogazioni in SQL

---

Fornire l'elenco dei veicoli del proprietario con codice fiscale 'RSSBNA71G02A403J' specificando la targa, il nome della categoria ed il nome del modello:

```
SELECT v.targa "Targa", m.nome "Nome Modello", c.nome "Nome categoria"  
FROM veicoli v, modelli m, categorie c  
WHERE v.cod_modello = m.cod_modello AND  
        v.cod_categoria = c.cod_categoria AND  
        v.targa IN  
            (SELECT targa  
             FROM proprieta  
             WHERE codice_fiscale = 'RSSBNA71G02A403J')
```

## Interrogazioni in SQL

---

Fornire l'elenco dei veicoli del proprietario con codice fiscale 'RSSBNA71G02A403J' specificando la targa, il nome della categoria ed il nome del modello

### JOIN Esplicito:

```
SELECT v.targa "Targa", m.nome "Nome Modello",  
        c.nome "Nome Categoria"  
FROM veicoli v [INNER] JOIN modelli m ON v.cod_modello = m.cod_modello  
        [INNER] JOIN categorie c ON v.cod_categoria = c.cod_categoria  
WHERE v.targa IN  
        (SELECT targa  
         FROM proprieta  
         WHERE codice_fiscale = 'RSSBNA71G02A403J')
```

### NOTA:

[INNER] si puo' omettere

## Interrogazioni in SQL

---

Fornire l'elenco dei veicoli **attualmente** in possesso del proprietario con codice fiscale 'RSSBNA71G02A403J' specificando la targa, il nome della categoria ed il nome del modello

```
SELECT v.targa "Targa", c.nome "Nome Categoria ", m.nome "Nome Modello"  
FROM veicoli v, modelli m, categorie c  
WHERE v.cod_modello = m.cod_modello  
AND v.cod_categoria = c.cod_categoria  
AND v.targa IN  
    (SELECT targa  
     FROM proprieta  
     WHERE codice_fiscale = 'RSSBNA71G02A403J' AND  
     data_cessione IS NULL)
```

## Interrogazioni in SQL

---

Fornire l'elenco dei veicoli **attualmente** in possesso del proprietario con codice fiscale 'RSSBNA71G02A403J' specificando la targa, il nome della categoria ed il nome del modello

```
SELECT v.targa "Targa", c.nome "Nome Categoria ", m.nome "Nome Modello"  
FROM veicoli v JOIN modelli m ON v.cod_modello = m.cod_modello  
JOIN categorie c ON v.cod_categoria = c.cod_categoria  
WHERE v.targa IN  
    (SELECT targa  
    FROM proprieta  
    WHERE codice_fiscale = 'RSSBNA71G02A403J' AND  
    data_cessione IS NULL)
```

### Join esterno **OUTER JOIN**:

Il join esterno estende, con valori nulli, le tuple che verrebbero tagliate fuori da un join (interno).

Tre versioni:

- **LEFT (sinistro)**: mantiene tutte le tuple del primo operando, estendendole con valori nulli, se necessario
- **RIGHT (destro)**: mantiene tutte le tuple del secondo operando
- **FULL (completo)**: mantiene tutte le tuple di entrambi gli operandi