

SCHEMA RELAZIONALE:

Proprietario (CodF, Nome, Cognome, Residenza)

Assicurazione (CodAss, Nome, Sede)

Sinistro (CodS, Localita, Data)

Auto (Targa, Marca, Cilindrata, Potenza, CodF, CodAss*)*

Autocoinvolte (CodS, Targa*, ImportoDelDanno)*

Prima cosa: creo tabelle

```
CREATE TABLE Proprietario(  
    CodF INT AUTO INCREMENT PRIMARY KEY, // 1 2 ... 100 (numero  
    progressivo)  
    // il nome è una stringa  
    // usiamo VARCHAR = VARIABILE CARATTERE(NUMERO CARATTERI)  
    Nome VARCHAR(50),  
    Cognome VARCHAR(50),  
    Residenza VARCHAR(100)  
);
```

Assicurazione e Sinistro sono uguali!

```
CREATE TABLE Assicurazione(  
    CodAss INT AUTO_INCREMENT PRIMARY KEY,  
    Nome VARCHAR(50),  
    Sede VARCHAR(50)  
);
```

```
CREATE TABLE Sinistro(  
    CodS INT AUTO_INCREMENT PRIMARY KEY,  
    Localita VARCHAR(100),  
    Descrizione VARCHAR(200),  
    Data DATE,  
    Orario TIME  
);
```

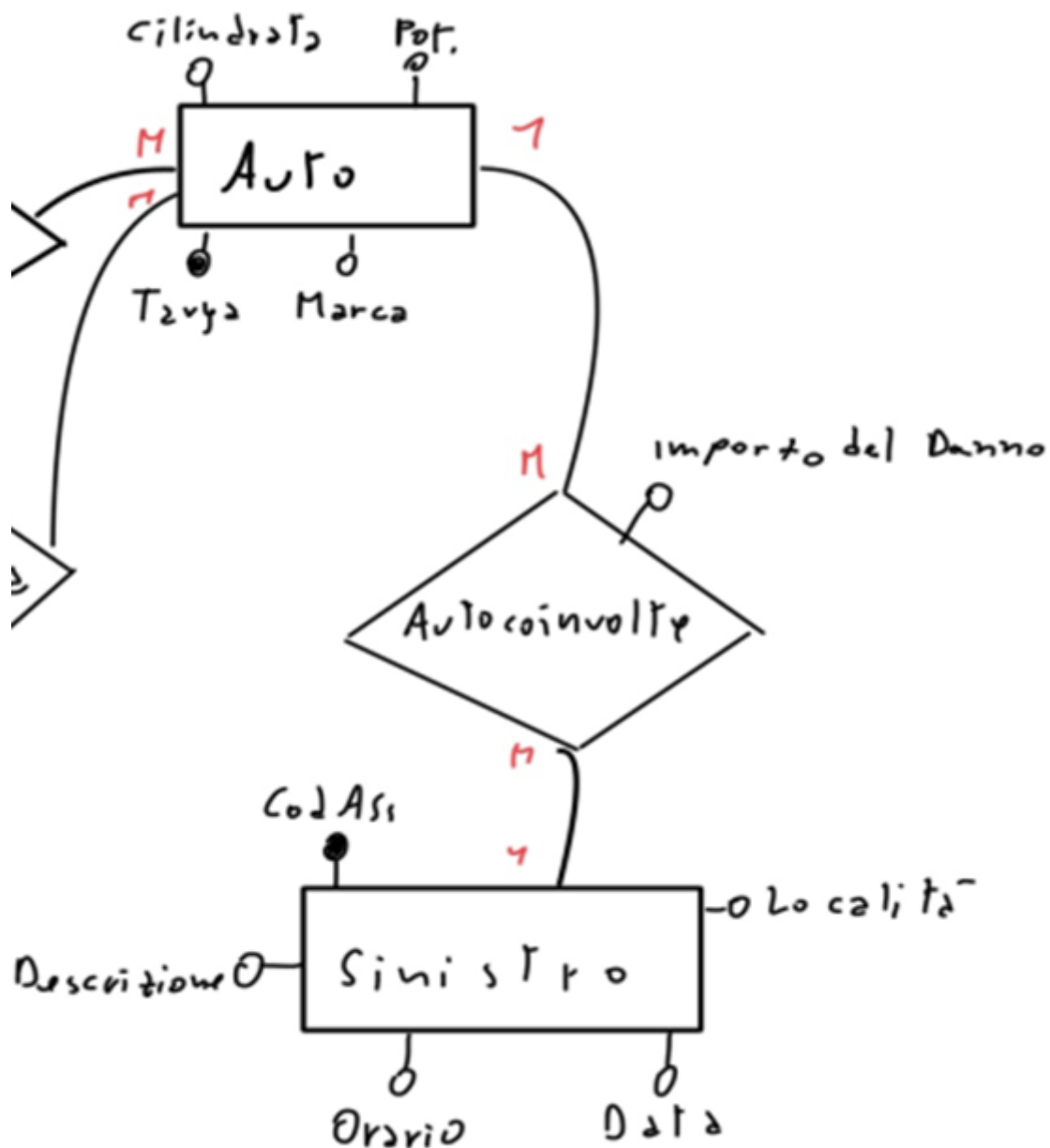
Auto (Targa, Marca, Cilindrata, Potenza, CodF, CodAss*)*

Codf e Codass sono chiavi esterne! --> FOREIGN KEY

A cascata = se cancello/aggiorno un campo in una relazione, siccome sono collegati da una relazione, le modifiche si ripercuotono a catena sul resto!

```
CREATE TABLE Auto(  
    Targa      VARCHAR(7) NOT NULL PRIMARY KEY,  
    Marca      VARCHAR(100),  
    Cilindrata INT,  
    Potenza    INT,  
    CodF       INT,  
    CodAss     INT,  
    FOREIGN KEY(CodF) REFERENCES Proprietario(CodF)  
        ON UPDATE CASCADE ON DELETE CASCADE,  
    FOREIGN KEY(CodAss) REFERENCES Assicurazione(CodAss)  
        ON UPDATE CASCADE ON DELETE CASCADE  
);
```

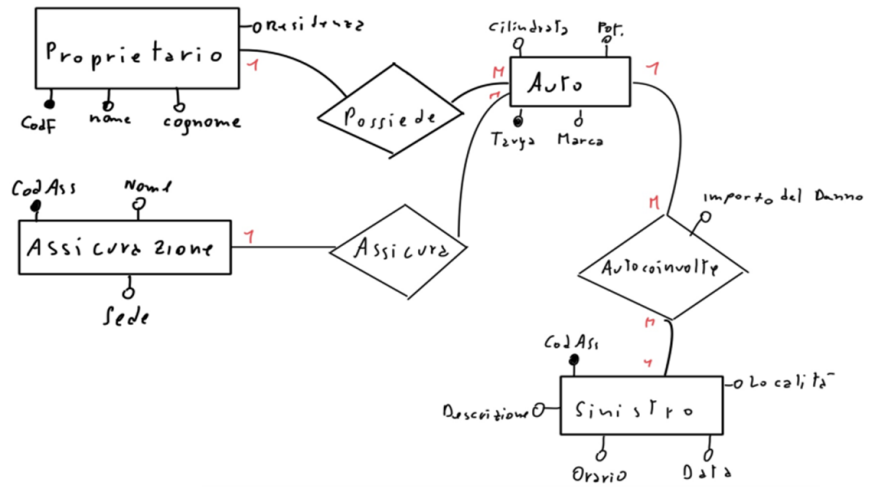
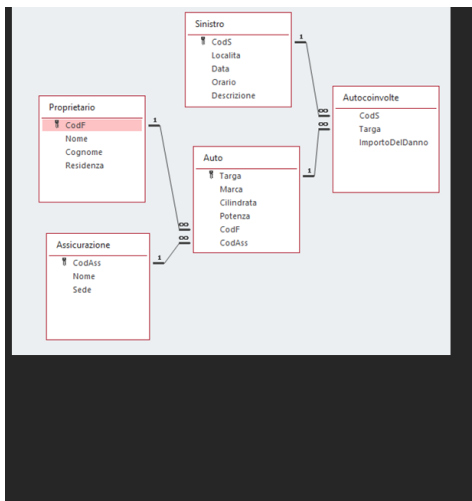
Attenzione: le relazioni possono diventare entità!



Qua diciamo:

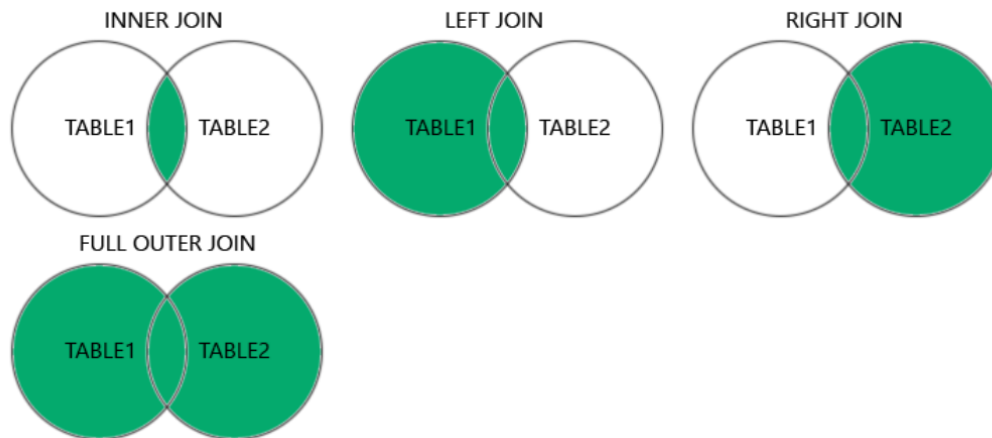
- un'auto è stata coinvolta in più sinistri con un importo del danno
- Viceversa:
- un sinistro ha coinvolto più auto con un importo del danno

Fino ad ora:



Relazioni - JOIN (Mettere insieme e capire da dove arrivano i dati di output = che mi interessano)

- **(INNER) JOIN** : Returns records that have matching values in both tables
- **LEFT (OUTER) JOIN** : Returns all records from the left table, and the matched records from the right table
- **RIGHT (OUTER) JOIN** : Returns all records from the right table, and the matched records from the left table
- **FULL (OUTER) JOIN** : Returns all records when there is a match in either left or right table



Query Auto

Tabelle:

- Auto (Targa, Marca, Cilindrata, Potenza, CodF)
- Proprietari (CodF, Nome)
- Assicurazioni (CodAss, Nome, Sede)
- Sinistri (CodS, Data, Targa, CodAss, Danni)

Query:

1- Targa e Marca delle Auto di cilindrata superiore a 2000 cc o di potenza superiore a 120 CV

- 2- Nome del proprietario e Targa delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV
- 3- Targa e Nome del proprietario delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV, assicurate presso la "SARA"
- 4- Targa e Nome del proprietario delle Auto assicurate presso la "SARA" e coinvolte in sinistri il 20/01/2002
- 5- Per ciascuna Assicurazione, il nome, la sede ed il numero di auto assicurate
- 6- Per ciascuna auto "Fiat", la targa dell'auto ed il numero di sinistri in cui è stata coinvolta
- 7- Per ciascuna auto coinvolta in più di un sinistro, la targa dell'auto, il nome dell' Assicurazione ed il totale dei danni riportati
- 8- CodF e Nome di coloro che possiedono più di un'auto
- 9- La targa delle auto che non sono state coinvolte in sinistri dopo il 20/01/2001
- 10- Il codice dei sinistri in cui non sono state coinvolte auto con cilindrata inferiore a 2000 cc

Ecco le soluzioni:

1.

```
SELECT Targa, Marca
FROM Auto
WHERE Cilindrata > 2000 OR Potenza > 120;
```

2.

```
SELECT p.Nome, a.Targa
FROM Auto a
JOIN Proprietari p ON a.CodF = p.CodF
WHERE a.Cilindrata > 2000 OR a.Potenza > 120;
```

3.

```
SELECT a.Targa, p.Nome
FROM Auto a
JOIN Proprietari p ON a.CodF = p.CodF
JOIN Assicurazioni ass ON a.CodAss = ass.CodAss
```

```
WHERE (a.Cilindrata > 2000 OR a.Potenza > 120)
AND ass.Nome = 'SARA';
```

4.

```
SELECT a.Targa, p.Nome
FROM Auto a
JOIN Proprietari p ON a.CodF = p.CodF
JOIN Sinistri s ON a.Targa = s.Targa
JOIN Assicurazioni ass ON s.CodAss = ass.CodAss
WHERE ass.Nome = 'SARA'
AND s.Data = '2002-01-20';
```

5.

```
SELECT ass.Nome, ass.Sede, COUNT(s.Targa) as NumeroAuto
FROM Assicurazioni ass
JOIN Sinistri s ON ass.CodAss = s.CodAss
GROUP BY ass.Nome, ass.Sede;
```

- Usiamo COUNT per contare le macchine coinvolte nei sinistri
- Quando usi COUNT (funzione di aggregazione), usi anche GROUP BY per prenderti le colonne usate in SELECT

6.

```
SELECT a.Targa, COUNT(s.CodS) as NumeroSinistri
FROM Auto a
LEFT JOIN Sinistri s ON a.Targa = s.Targa
WHERE a.Marca = 'FIAT'
GROUP BY a.Targa;
```

7.

```
SELECT a.Targa, ass.Nome, SUM(s.Danni) as TotaleDanni
FROM Auto a
JOIN Sinistri s ON a.Targa = s.Targa
JOIN Assicurazioni ass ON s.CodAss = ass.CodAss
GROUP BY a.Targa, ass.Nome
HAVING COUNT(s.CodS) > 1;
```

8.

```
SELECT p.CodF, p.Nome
FROM Proprietari p
JOIN Auto a ON p.CodF = a.CodF
GROUP BY p.CodF, p.Nome
HAVING COUNT(a.Targa) > 1;
```

9.

```
SELECT a.Targa
FROM Auto a
WHERE a.Targa NOT IN (
    SELECT s.Targa
    FROM Sinistri s
    WHERE s.Data > '2001-01-20'
);
```

10.

```
SELECT s.CodS
FROM Sinistri s
WHERE NOT EXISTS (
    SELECT *
    FROM Auto a
    WHERE a.Targa = s.Targa
    AND a.Cilindrata < 2000
);
```