

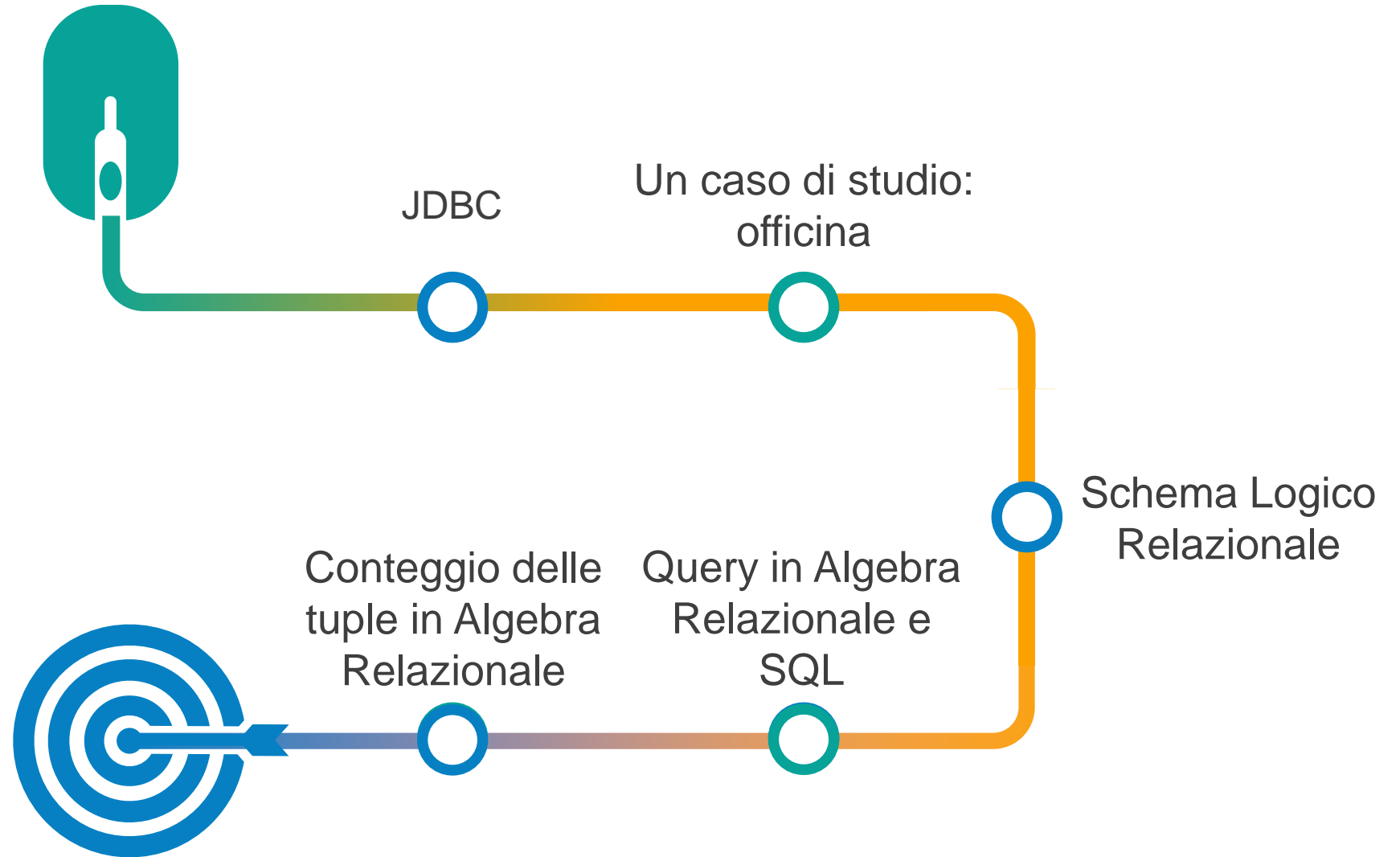


BASI DI DATI QUERY

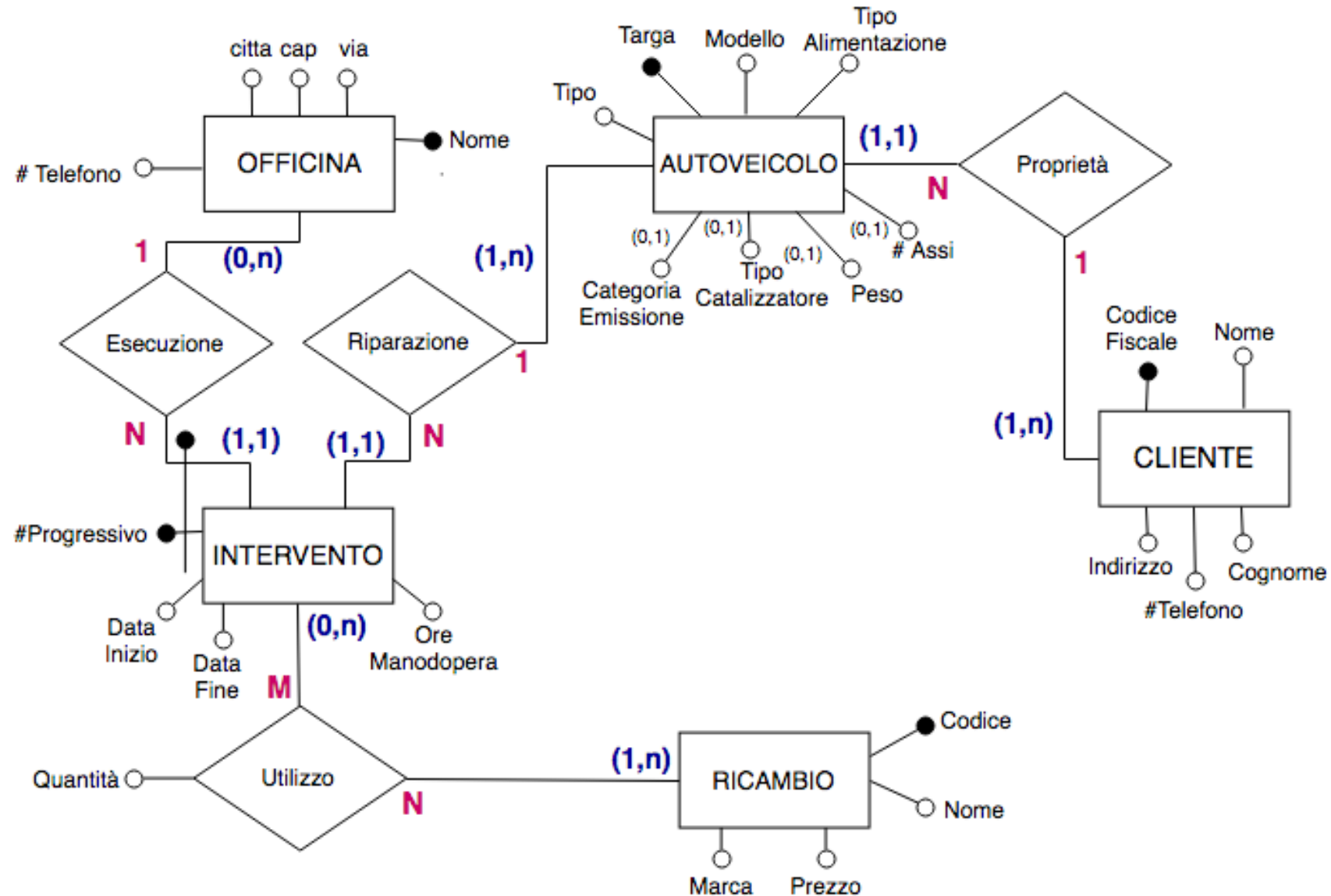
Polese G. Caruccio L. Breve B.

a.a. 2023/2024

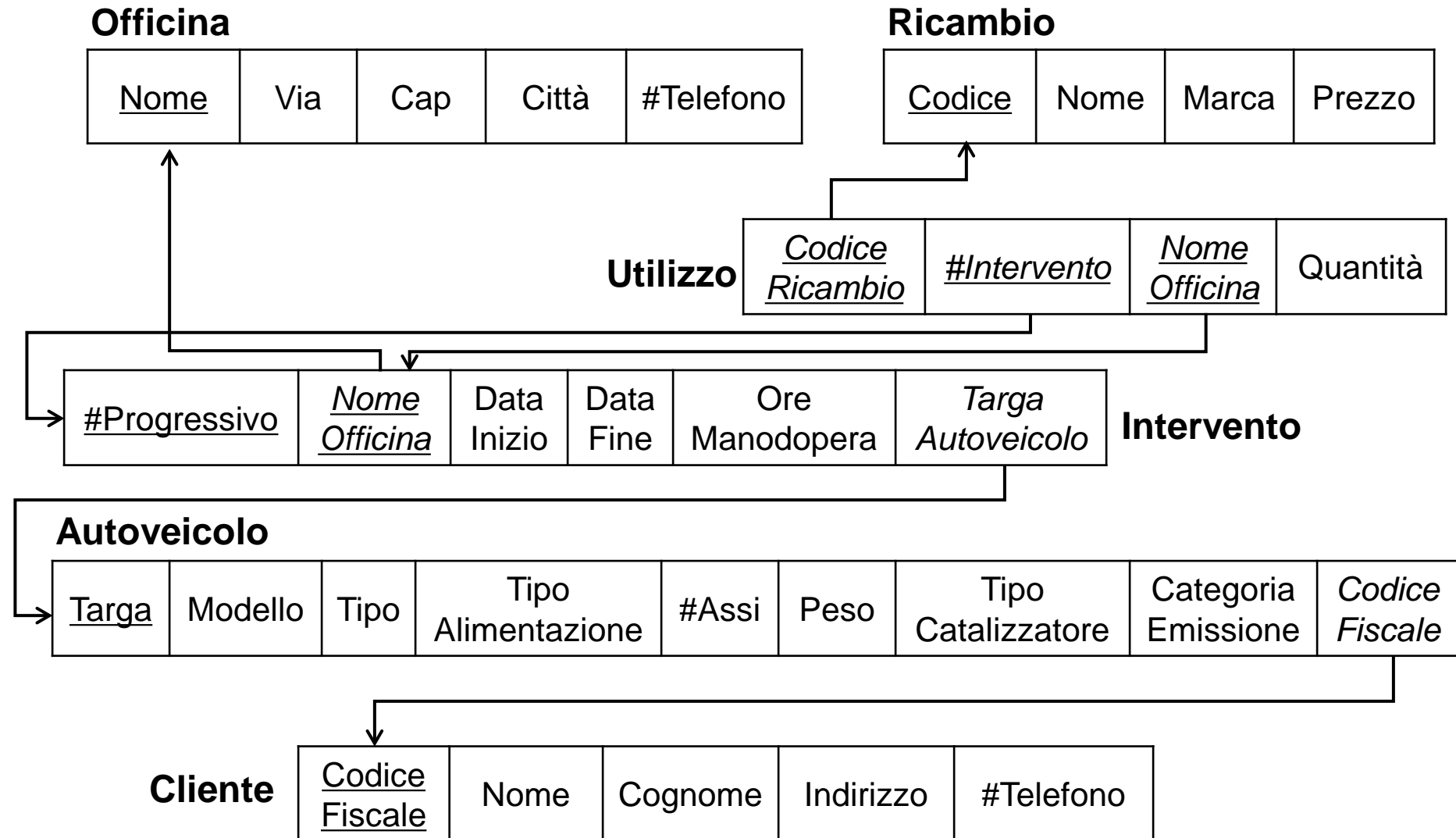
Outline



SCHEMA RISTRUTTURATO



MAPPING NELLO SCHEMA LOGICO RELAZIONALE



INSERIMENTO DEI DATI

- Supponiamo di voler inserire i seguenti dati

officina

nome	via	cap	città	#telefono
Officina F.lli Ferraro	Lungomare C. Colombo, 307	84129	Salerno	NULL
Officina Ferraro Giuseppe	Via Madonna di Fatima, 130	84129	Salerno	NULL
Meccanica Ferraro A. & Figli	Via Venezia, 23	84098	Pontecagnano (SA)	089221154

ricambio

codice	nome	marca	prezzo
598382	Kit pastiglie freno, Freno a disco	Valeo	90,65
CT704	Cinghia dentata	CONTITECH	56,67
0451103351	Filtro olio	BOSCH	17,93

INSERIMENTO DEI DATI

- Supponiamo di voler inserire i seguenti dati

autoveicolo

targa	modello	tipo alimen.	tipo	#assi	peso	tipo catal.	Cat. Emis.	Codice Fiscale
FF111	Opel Corsa	GPL	2	NULL	NULL	A	Euro 6	LNEBCH05B41G230M
AB123	Fiat Punto	Benzina	2	NULL	NULL	B	Euro 4	MRARSS60D63A717X
CA171	Eurocarco 190E30	Diesel	1	4X2	9 T	NULL	NULL	LNEBCH05B41G230M

cliente

codice fiscale	nome	cognome	città	telefono
LNEBCH05B41G230M	Elena	Bianchi	Via Napoli ,15 Pagani	NULL
MRARSS60D63A717X	Maria	Rossi	Via Ripa, 128 Battipaglia	0828444444
STFVRD99R07H703G	Stefano	Verdi	P.zza XXIV Maggio 3, Salerno	089151515

CARICAMENTO DEI DATI

- Supponiamo di voler inserire i seguenti dati

intervento

#prog	nome officina	data inizio	data fine	ore manodopera	targa autoveicolo
1	Meccanica Ferraro A. & Figli	2018-09-13	2018-09-14	5	CA171
1	Officina Ferraro Giuseppe	2018-10-22	2018-10-24	10	FF111
2	Meccanica Ferraro A. & Figli	2018-12-03	2018-12-03	2	CA171

utilizzo

codice ricambio	#intervento	nome officina	quantità
598382	1	Meccanica Ferraro A. & Figli	2
CT704	1	Officina Ferraro Giuseppe	1
0451103351	2	Meccanica Ferraro A. & Figli	1
CT704	1	Meccanica Ferraro A. & Figli	1

QUERY DI INTERROGAZIONE: ALGEBRA RELAZIONALE

- La proiezione

`PROJ lista_attributi(tabella)`

- Esempio

- Restituire il nome e il cognome dei clienti

PROJ Nome, Cognome (Cliente)

QUERY DI INTERROGAZIONE: SQL

- Il comando select

```
SELECT lista_elementi_selezione  
FROM lista_riferimenti_tabella
```

- Esempio

- Restituire il nome e il cognome dei clienti

```
SELECT nome, cognome  
FROM cliente
```

QUERY DI INTERROGAZIONE: SQL

- Il comando select

```
SELECT [ALL|DISTINCT] lista_elementi_selezione  
FROM lista_riferimenti_tabella
```

- Esempio

- Restituire il cognome dei clienti

```
SELECT cognome  
FROM cliente
```

```
SELECT DISTINCT cognome  
FROM cliente
```

- NOTA: In Algebra Relazionale la proiezione elimina le tuple duplicate

QUERY DI INTERROGAZIONE: SQL

- Il comando select

```
SELECT [ALL|DISTINCT] lista_elementi_selezione  
FROM lista_riferimenti_tabella
```

- Esempio

- Restituire tutti i dati delle officine

```
SELECT *  
FROM officina
```

QUERY DI INTERROGAZIONE: ALGEBRA RELAZIONALE

- Il comando select

```
SELECT lista_condizioni(tabella)
```

- Esempio

- Restituire tutte le officine di Salerno

```
SEL_città='Salerno' (Officina)
```

JOIN NATURALE: ALGEBRA RELAZIONALE

- Join naturale

`relazione_1 JOIN relazione_2`

- Esempio

- Restituire tutti i clienti (nome e cognome) proprietari di autoveicoli con categoria emissione "Euro 6"

PROJ_{Nome, Cognome}(**Cliente**
JOIN

(**SEL**_{Categoria Emissione='Euro 6'}(**Autoveicolo**))

JOIN NATURALE: SQL

- Join naturale

```
SELECT [ALL|DISTINCT] lista_elementi_selezione  
      FROM tabella_1 NATURAL JOIN tabella_2  
      [WHERE espressione_condizionale]
```

- Esempio

- Restituire tutti i clienti (nome e cognome) proprietari di autoveicoli con categoria emissione "Euro 6"

```
SELECT nome, cognome  
FROM cliente NATURAL JOIN autoveicolo  
WHERE categoriaEmissione = 'Euro 6'
```

EQUI-JOIN: ALGEBRA RELAZIONALE

- Equi-Join

`relazione_1 JOIN condizione join relazione_2`

- Esempio

- Restituire tutte le auto (targa, modello) su cui sono stati effettuati interventi a partire dal 03/11/2018

PROJ_{Targa,Modello}(Autoveicolo
JOIN_{Targa = Targa Autoveicolo}
(**SEL**_{Data Inizio} ≥ '2018-11-03' (Intervento)))

EQUI-JOIN: SQL

- Equi-Join

```
SELECT [ALL| DISTINCT] lista_elementi_selezione
      FROM tabella_1 INNER JOIN tabella_2
      ON condizione_join
      [WHERE espressione_condizionale]
```

- Esempio

- Restituire tutte le auto (targa, modello) su cui sono stati effettuati interventi a partire dal 03/11/2018

SELECT targa, modello

FROM autoveicolo **INNER JOIN** intervento **ON**
autoveicolo.targa = intervento.targaAutoveicolo

WHERE dataInizio >= '2018-11-03'

FUNZIONI DI AGGREGAZIONE: SQL

- Esempio

- Per ogni ricambio restituire il numero di interventi in cui sono stati impiegati (nome,numInterventi)

```
SELECT nome, COUNT(codiceRicambio) AS numInterventi
FROM ricambio INNER JOIN utilizzo
  ON ricambio.codice = utilizzo.codiceRicambio
GROUP BY codiceRicambio, nome
```

FUNZIONI DI AGGREGAZIONE: SQL

- Esempio

- Per ogni ricambio restituire il numero di interventi in cui sono stati impiegati e la quantità totale richiesta (nome, numInterventi, quantitaTotale)

```
SELECT nome, COUNT(codiceRicambio) AS numInterventi,  
SUM(quantità) AS quantitaTotale
```

```
FROM ricambio INNER JOIN utilizzo ON ricambio.codice =  
utilizzo.codiceRicambio
```

```
GROUP BY codiceRicambio,nome,quantità
```

FUNZIONI DI AGGREGAZIONE: SQL

- Esempio

- Per ogni ricambio restituire il numero di interventi in cui sono stati impiegati, con una quantità maggiore di uno, e la quantità totale richiesta (nome, numInterventi, quantitaTotale)

```
SELECT nome, COUNT(codiceRicambio) AS numInterventi,  
        SUM(quantità) AS quantitaTotale
```

```
FROM ricambio INNER JOIN utilizzo ON ricambio.codice =  
        utilizzo.codiceRicambio
```

```
WHERE quantità > 1
```

```
GROUP BY codiceRicambio,nome,quantità
```

FUNZIONI DI AGGREGAZIONE: SQL

- Esempio

- Per ogni ricambio restituire la quantità totale richiesta se e solo se sono stati impiegati in più di un intervento (nome, quantitaTotale)

```
SELECT nome, SUM(quantità) AS quantitaTotale
FROM ricambio INNER JOIN utilizzo ON ricambio.codice =
    utilizzo.codiceRicambio
GROUP BY codiceRicambio,nome
HAVING COUNT(codiceRicambio) > 1
```

QUERY ANNIDATE: SQL

- Esempio
 - Restituire i dati dei clienti (nome, cognome) che hanno richiesto almeno un intervento

```
SELECT DISTINCT nome, cognome  
FROM cliente NATURAL JOIN autoveicolo INNER JOIN  
intervento ON autoveicolo.targa =  
intervento.targaAutoveicolo
```


QUERY ANNIDATE: SQL

- Esempio
 - Restituire i dati dei clienti (nome, cognome) che hanno richiesto almeno un intervento

```
SELECT nome, cognome
FROM cliente AS c
WHERE EXISTS (SELECT *
              FROM autoveicolo AS a INNER JOIN
                intervento AS i
                ON a.targa = i.targaAutoveicolo
              WHERE a.codiceFiscale=c.codiceFiscale)
```

QUERY ANNIDATE: SQL

- Esempio

- Restituire i dati dell'officina (nome) con il massimo numero di ore di manodopera impiegate in un qualche intervento

```
SELECT nome
FROM officina AS o INNER JOIN intervento AS i
    ON o.nome = i.nomeOfficina
WHERE oreManodopera >= (
    SELECT MAX(oreManodopera)
    FROM intervento)
```


CONTEGGIO TUPLE ALGEBRA RELAZIONALE

- Selezionare i clienti (nome, cognome) che abbiano autoveicoli tutti dello stesso tipo

```
PROJnome, cognome(Cliente)
-
( PROJnome, cognome (Cliente JOIN
  (SELTipo<>Tipo1 (Autoveicolo
    JOIN
      (RENtipo1←tipo( PROJcodiceFiscale, tipo(Autoveicolo))))))
```

QUERY IN SQL

- Selezionare i clienti (nome, cognome) che abbiano autoveicoli tutti dello stesso tipo

```
SELECT nome, cognome  
FROM cliente NATURAL JOIN autoveicolo  
GROUP BY codiceFiscale, nome, cognome  
HAVING COUNT(DISTINCT tipo) = 1
```

CONTEGGIO TUPLE ALGEBRA RELAZIONALE

- Selezionare i ricambi (nome, marca, prezzo) che siano stati utilizzati in almeno 2 interventi

```
(PROJnome,marca,prezzo (Ricambio JOIN  
(SEL#intervento<>#intervento1 (Utilizzo  
JOIN  
(REN#intervento1<#intervento  
PROJcodiceRicambio,#intervento(Utilizzo))))))
```

QUERY IN SQL

- Selezionare i ricambi (nome, marca, prezzo) che siano stati utilizzati in almeno 2 interventi

```
SELECT nome, marca, prezzo
FROM ricambio JOIN utilizzo ON
    ricambio.Codice = utilizzo.CodiceRicambio
GROUP BY CodiceRicambio, nome, marca, prezzo
HAVING COUNT(#intervento) > 1
```