# Esercitazione fine modulo SQL

Vincenzo Giacalone Biondo

## Descrizione

Ho ipotizzato un caso studio riguardante un’analisi sul mercato automobili.

L’analisi ipotetica riguarda:

* Le case automobilistiche,
* I singoli modelli auto prodotti dalle case automobilistiche,
* Le concessionarie nelle quali questi modelli auto vengono vendute,
* I venditori impiegati nelle concessionarie,
* Gli acquirenti.

Si è proceduto alla realizzazione del modello ER. Sono state individuate le seguenti entità:

* **Case automobilistiche** e **automobili**, collegati dalla relazione “Produzione”. La relazione ha molteplicità 1,n nel verso case automobilistiche->automobili e 1,1 nel senso contrario, in quanto un gruppo automobilistico non ha un limite di auto prodotte, mentre ciascuna auto è prodotta da un singolo gruppo,
* **Concessionaria** e **venditore,** collegati dalla relazione ‘impiego’. La relazione ha molteplicità 1,n nel verso concessionaria -> venditore e 1,1 nel senso opposto, in quanto ogni concessionaria può avere un numero indefinito di venditori, ma ogni venditore è necessariamente assegnato ad una concessionaria. Tra le due entità è presente un ulteriore relazione di molteplicità 1,1 nel senso concessionaria -> venditore e 0,1 nel senso opposto che riguarda la definizione del venditore referente per ciascuna concessionaria.
* **Cliente**, entità che riguarda l’anagrafica degli acquirenti auto. L’entità è collegata tramite la relazione ‘Acquisti’ alle entità venditore e automobili.

La molteplicità della relazione è 1,1 con l’entità ‘automobili’, mentre 1,n verso le altre due entità, in quanto ogni automobile può essere venduta una sola volta da un solo venditore ad un solo acquirente, mentre non c’è limite al numero di auto acquistabili dai clienti e vendibili dai venditori.

* **Nazioni**, entità che raggruppa le ragioni all’interno delle nazioni.

Nel passaggio al modello logico, la relazione di acquisto diviene un’entità a sé stante vista la molteplicità di tale relazione.

Per le entità nelle quali era possibile identificare una chiave primaria univoca, questa è stata selezionata. Per le entità nelle quali ciò non era possibile, è stata ideata una chiave primaria ‘ID’ o ‘numero di protocollo’ univoca.

In seguito è stato ipotizzato un modello fisico.

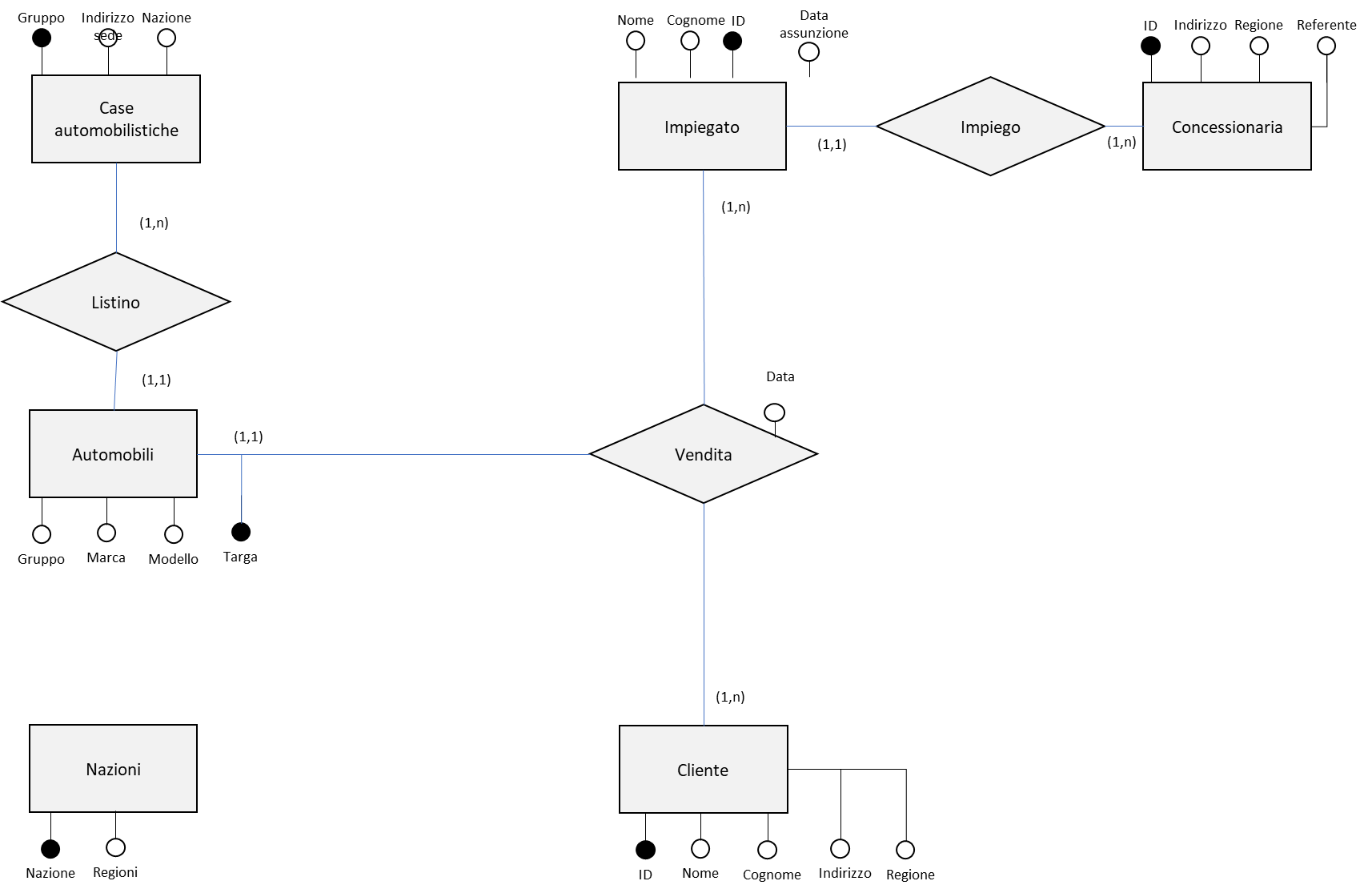
Sono state dapprima create le tabelle, in seguito sono state realizzate le seguenti viste:

* **Best sellers**, con l’obbiettivo di individuare le auto più vendute in assoluto,
* **Venditore del mese**, con l’obbiettivo di individuare i 3 venditori più attivi negli ultimi 31 giorni,
* **Biggest company**¸ con l’obbiettivo di individuare i gruppi auto con più vendite attive nell’ultimo anno.

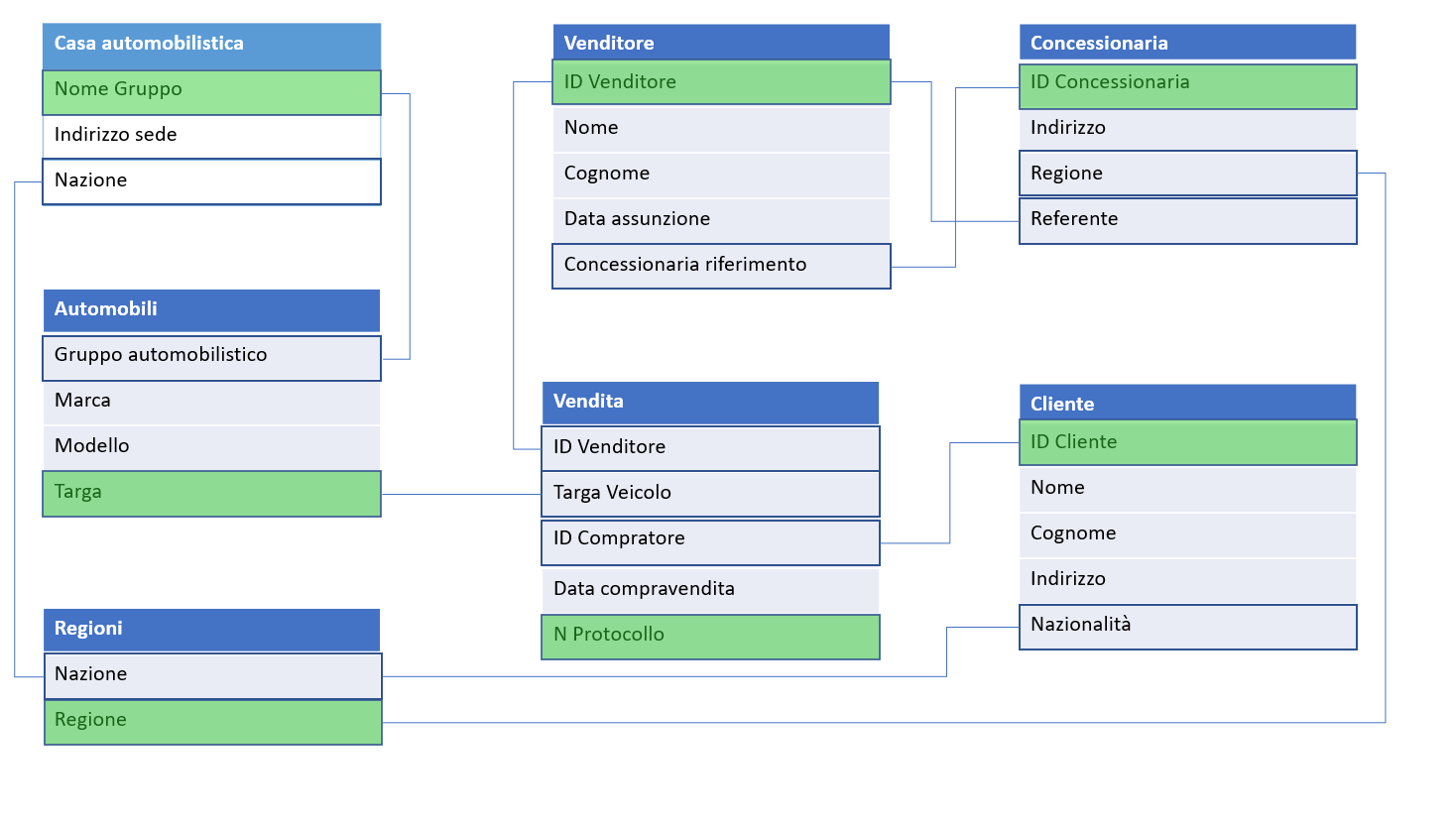
Oltre alle viste, sono state ipotizzate le seguenti quary:

* Le nazioni nelle quali vengono vendute più auto,
* I venditori con più anni di carriera alle spalle,
* Il numero di vendite per concessionaria,
* I dati dei referenti delle concessionarie che vendono meno auto,
* Le nazioni che producono più auto vendute,
* La nazionalità dei clienti che acquistano auto italiane,
* I dati dei clienti che hanno acquistato fiat 500 o fiat tipo dopo il 2020.

## Modello ER



## Modello logico



## Modello fisico

USE database;

CREATE TABLE case\_auto (

gruppo varchar(64) NOT NULL,

indirizzo varchar(64),

città varchar(64),

nazione varchar(64),

PRIMARY KEY (gruppo)

);

CREATE TABLE automobili (

gruppo varchar(64) NOT NULL,

marca varchar(64),

modello varchar(64),

targa varchar(8) NOT NULL,

PRIMARY KEY (targa),

FOREIGN KEY (gruppo) REFERENCES case\_auto(gruppo)

);

CREATE TABLE venditore (

ID int NOT NULL AUTO\_INCREMENT,

nome varchar(64),

cognome varchar(64),

data\_assunzione date,

concessionaria varchar(64) NOT NULL,

PRIMARY KEY (ID)

);

CREATE TABLE concessionarie (

ID int NOT NULL AUTO\_INCREMENT,

nome varchar(64),

indirizzo varchar(64),

regione varchar(64) NOT NULL,

referente int NOT NULL,

PRIMARY KEY (ID),

foreign key (referente) references venditore(ID)

);

CREATE TABLE clienti (

ID int NOT NULL AUTO\_INCREMENT,

nome varchar(64),

cognome varchar(64),

Indirizzo varchar(64),

nazionalità varchar(64) NOT NULL,

PRIMARY KEY (ID)

);

CREATE TABLE vendite (

n int NOT NULL AUTO\_INCREMENT,

id\_venditore int NOT NULL,

targa varchar(64) NOT NULL,

id\_cliente int NOT NULL,

data\_c date,

PRIMARY KEY (n),

foreign key (id\_venditore) references venditore(ID),

foreign key (id\_cliente) references clienti(ID),

foreign key (targa) references automobili(targa)

);

CREATE TABLE regioni (

regione varchar(64) NOT NULL,

nazione varchar(64) NOT NULL,

PRIMARY KEY (regione)

);

CREATE VIEW best\_sellers AS

SELECT automobili.marca, automobili.modello,

count(\*) AS n\_vendite

FROM automobili RIGHT JOIN vendite ON automobili.targa = vendite.targa

group by automobili.marca, automobili.modello

ORDER BY n\_vendite

LIMIT 10;

CREATE VIEW venditore\_del\_mese AS

SELECT s.nome, s.cognome, count(\*) AS n\_vendite

FROM venditore AS s RIGHT JOIN vendite AS v ON s.ID = v.ID\_venditore

WHERE CURDATE() - v.data\_c < 32

group by s.nome, s.cognome

ORDER BY n\_vendite

LIMIT 3;

CREATE VIEW biggest\_companies\_currentyear AS

SELECT g.gruppo, count(\*) AS vendite\_totali

FROM case\_auto AS g LEFT JOIN automobili AS a ON g.gruppo = a.gruppo LEFT JOIN vendite AS v ON a.targa = v.targa

WHERE CURDATE() - v.data\_c < 366

GROUP BY g.gruppo

ORDER BY vendite\_totali;

SELECT g.nazione, count(\*) AS n\_vendite

FROM regioni as g RIGHT JOIN concessionarie as c ON g.regione = c.regione RIGHT JOIN venditore AS s ON c.ID = s.concessionaria RIGHT JOIN vendite AS v ON s.ID = v.ID\_venditore

GROUP BY g.nazione

ORDER BY n\_vendite;

SELECT nome, cognome, CURDATE() - data\_assunzione AS anzianità

FROM venditore

Order by anzianità DESC

SELECT c.ID, c.nome, c.indirizzo, count(\*) AS vendite\_totali

FROM concessionarie AS c RIGHT JOIN venditore AS v ON c.ID = v.concessionaria RIGHT JOIN vendite AS s ON v.ID = s.ID\_venditore

GROUP BY c.ID

ORDER BY c.regione

SELECT n.nome, n.cognome, count(\*) AS vendite\_totali

FROM venditore AS n RIGHT JOIN concessionarie AS c ON n.ID = c.referente RIGHT JOIN venditore AS v ON c.ID = v.concessionaria RIGHT JOIN vendite AS s ON v.ID = s.ID\_venditore

Group BY c.ID

ORDER BY vendite\_totali ASC

SELECT g.nazione, count(\*) AS produzione\_auto\_vendute

FROM case\_auto AS g LEFT JOIN automobili AS a ON g.gruppo = a.gruppo LEFT JOIN vendite AS v ON a.targa = v.targa

GROUP BY g.nazione

SELECT c.nazionalità, count(\*) AS acquisiti\_ITA

FROM clienti AS c RIGHT JOIN vendite as v ON c.ID = v.id\_cliente right join automobili as a on v.targa = a.targa right join case\_auto as g on a.gruppo = g.gruppo

WHERE g.nazione = 'Italia'

GROUP BY c.nazionalità

SELECT c.nome, c.cognome, c.indirizzo

FROM clienti AS c RIGHT JOIN vendite as v ON c.ID = v.id\_cliente right join automobili as a on v.targa = a.targa

WHERE v.data\_c > '2020-01-01' AND a.marca = 'Fiat' AND a.modello = '500' OR a.modello = 'Tipo'

GROUP BY c.ID