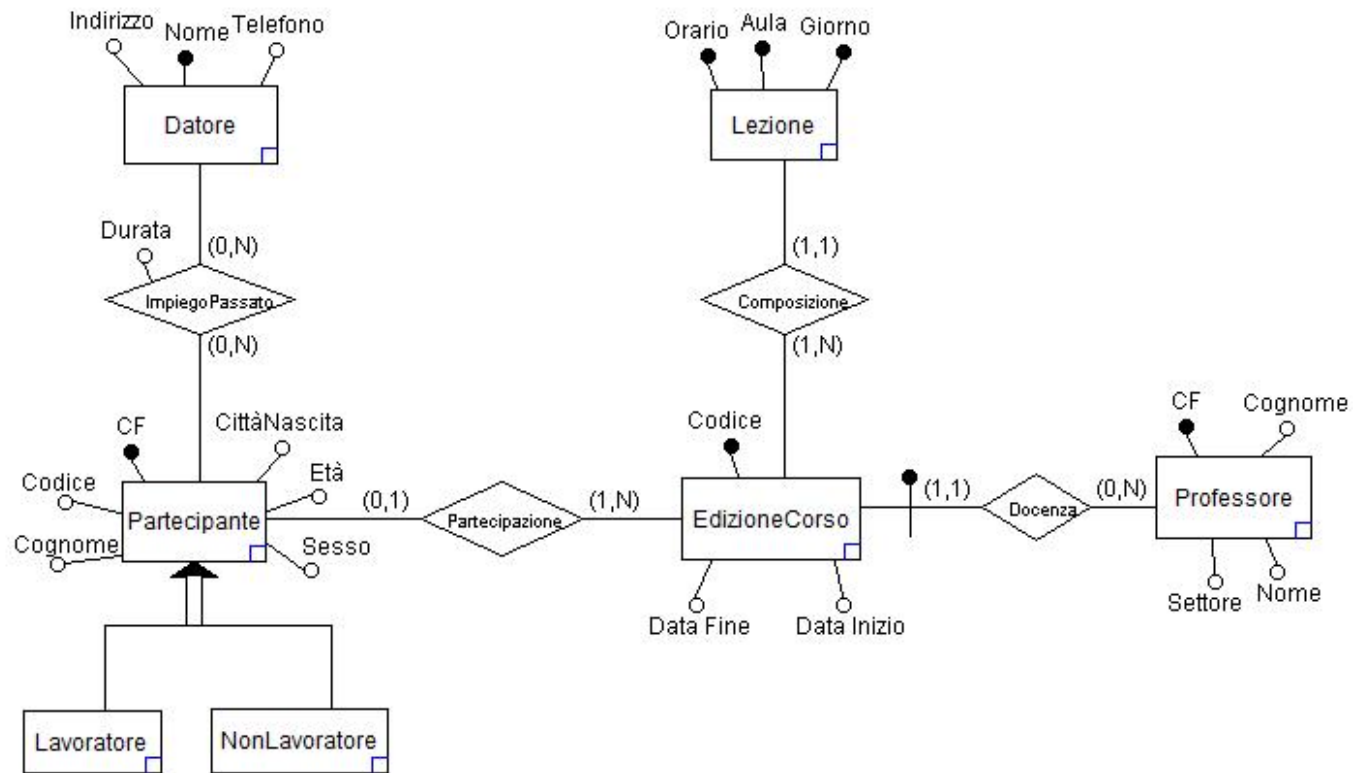


# Tutorato di Basi di Dati

Anno 2017/2018



Soluzione dei Classwork

# Outline

---

- ▶ Soluzione classwork – Tracce A e B
  - ▶ Schema concettuale del database
  - ▶ Ristrutturazione
    - ▶ Analisi delle ridondanze
    - ▶ Rimozione delle gerarchie
    - ▶ Schema Ristrutturato
  - ▶ Schema logico relazionale
  - ▶ Creazione del database mediante DDL di SQL

# Traccia A

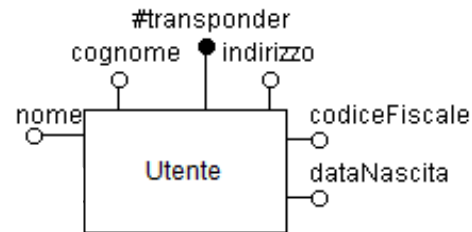
## Schema concettuale del database

Utilizzando il modello ER, progettare lo schema concettuale del database per la gestione degli accessi ai parcheggi riservati dell'Università. Esso dovrà memorizzare i dati degli utenti abilitati al parcheggio nelle aree riservate (nome, cognome, data di nascita, codice fiscale, indirizzo e numero di trasponder), distinguendo tra i dipendenti a tempo pieno ed i contrattisti. Per i primi occorrerà inoltre memorizzare la qualifica ed il dipartimento di appartenenza, mentre per i contrattisti occorrerà memorizzare il tipo di contratto (borsa di studio, assegno di ricerca, etc), la data di inizio e quella di conclusione. Solo i dipendenti ed i contrattisti sono abilitati al parcheggio, ma questi ultimi solo fino alla scadenza del proprio contratto. Per ogni utente abilitato occorre inoltre memorizzare le aree di parcheggio riservate cui il proprio trasponder consente di accedere. Infine, per ciascuna area di parcheggio occorre memorizzare il codice della sbarra di accesso, la superficie in metri quadrati, il numero di posti totali, quelli coperti, quelli occupati, quelli disponibili ed il numero di utenti abilitati al parcheggio.

# Traccia A

## Schema concettuale del database

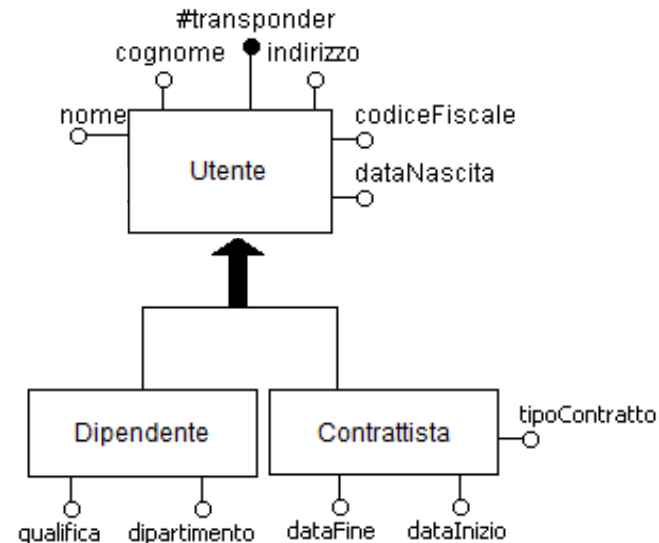
[...] Esso dovrà memorizzare i dati degli **utenti** abilitati al parcheggio nelle aree riservate (*nome, cognome, data di nascita, codice fiscale, indirizzo e numero di trasponder*), [...]



# Traccia A

## Schema concettuale del database

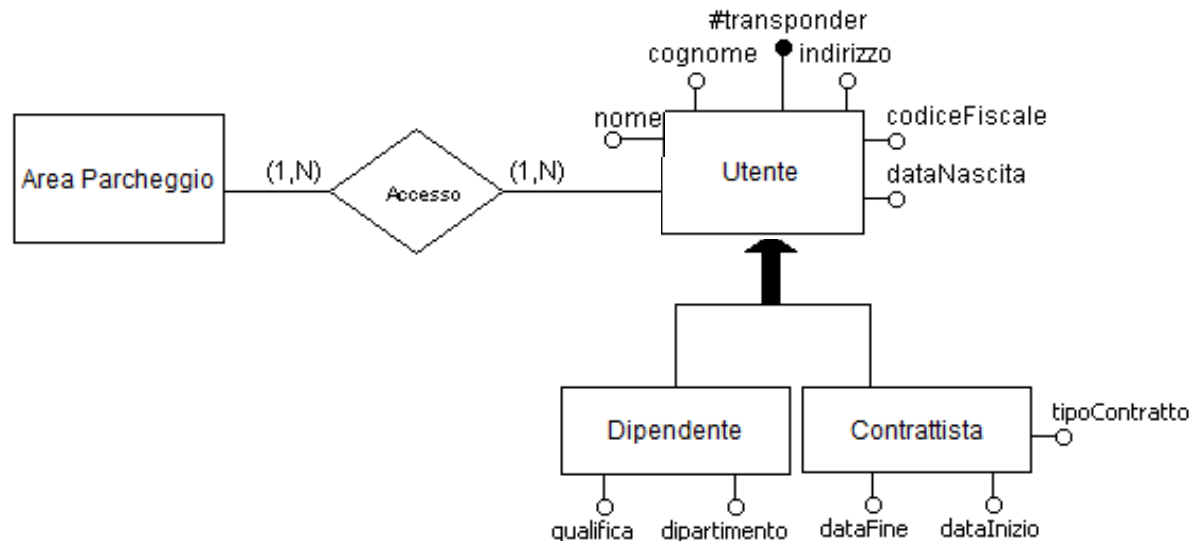
[...] distinguendo tra i **dipendenti** a tempo pieno ed i **contrattisti**. Per i primi occorrerà inoltre memorizzare la *qualifica* ed il *dipartimento* di appartenenza, mentre per i **contrattisti** occorrerà memorizzare il *tipo di contratto* (borsa di studio, assegno di ricerca, etc), la *data di inizio* e quella di *conclusione*. [...]



# Traccia A

## Schema concettuale del database

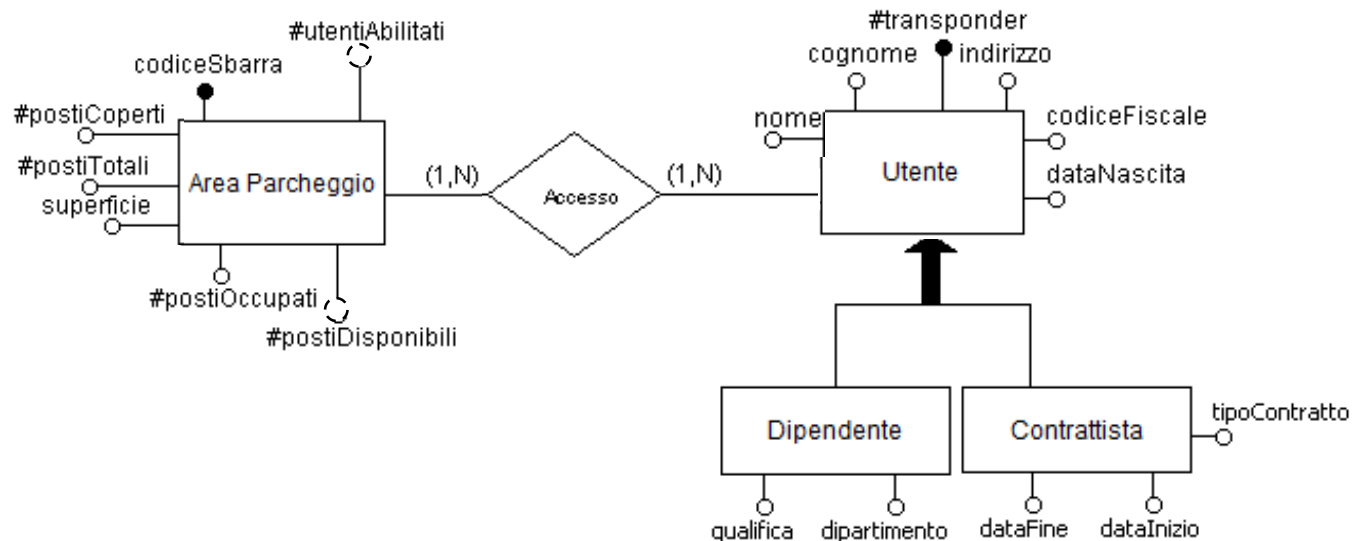
[...] Solo i dipendenti ed i contrattisti sono abilitati al parcheggio, ma questi ultimi solo fino alla scadenza del proprio contratto. Per ogni utente abilitato occorre inoltre memorizzare le **aree di parcheggio** riservate cui il proprio trasponder consente di **accedere**. [...]



# Traccia A

## Schema concettuale del database

[...] Infine, per ciascuna **area di parcheggio** occorre memorizzare il *codice della sbarra di accesso*, la *superficie in metri quadrati*, il *numero di posti totali*, quelli coperti, quelli occupati, quelli disponibili ed il *numero di utenti abilitati al parcheggio*.



# Traccia A

---

## Ristrutturazione

Calcolare il carico applicativo e ristrutturare lo schema ER prodotto, valutando se mantenere gli attributi ridondanti individuati. I dipendenti a tempo pieno sono circa 1.500 tra personale docente e non docente, i contrattisti sono circa 500, le aree di parcheggio 10. In media, i dipendenti possono accedere alla metà dei parcheggi riservati, mentre i contrattisti possono accedere ad un paio di essi. Le principali operazioni da eseguire su questo database sono 15, di cui le tre più frequenti sono:

OP1) Ingresso/uscita di un veicolo da un parcheggio riservato, (2.000 volte/giorno lavorativo).

OP2) Al termine di un giorno lavorativo, verifica scadenza di tutti i contratti, con cancellazione contrattisti il cui contratto è scaduto (in media ne scadono 2 per giorno lavorativo).

OP3) Stampa mensile di un report che mostri i dati di ciascun parcheggio, incluso il numero di utenti abilitati.



# Traccia A

## Carico Applicativo

- ▶ I dipendenti a tempo pieno sono circa 1.500 tra personale docente e non docente,
- ▶ i contrattisti sono circa 500,
- ▶ le aree di parcheggio 10.
  - ▶ In media, i dipendenti possono accedere alla metà dei parcheggi riservati, mentre
  - ▶ i contrattisti possono accedere ad un paio di essi.

### Tavola dei volumi

Concetto	Costrutto	Volume
AreaParcheggio	E	10
Accesso	R	8500
Utente	E	2000
Dipendente	SE	1500
Contrattista	SE	500

$$(1500 \times 5) + (500 \times 2)$$

# Traccia A

## Carico Applicativo

► Le principali operazioni da eseguire su questo database sono 15, di cui le tre più frequenti sono:

- OP1) Ingresso/uscita di un veicolo da un parcheggio riservato, (2.000 volte/giorno lavorativo).
- OP2) Al termine di un giorno lavorativo, verifica scadenza di tutti i contratti, con cancellazione contrattisti il cui contratto è scaduto (in media ne scadono 2 per giorno lavorativo).
- OP3) Stampa mensile di un report che mostri i dati di ciascun parcheggio, incluso il numero di utenti abilitati.

### Tavola delle operazioni

Operazione	Tipo	Frequenza
OP1	I	2000/giorno lavorativo
OP2	B	1/giorno lavorativo
OP3	B	1/mese

$(2000 \times 22) = 44.000/\text{mese}$

22/mese

# Traccia A

---

## Carico Applicativo

► Nota: lo schema ER contiene due attributi ridondanti sull'entità “Area Parcheggio”, tuttavia non è necessario analizzare gli accessi (con o senza ridondanza) per l'attributo *#postiDisponibili*, dato che tra le operazioni più frequenti solo

OP1 - Ingresso/uscita di un veicolo da un parcheggio riservato  
(2.000 volte/giorno lavorativo)

richiede l'aggiornamento di *#postiDisponibili*, ma tale aggiornamento può essere fatto insieme all'aggiornamento dell'attributo *#postiOccupati*.

Inoltre, gli accessi a tale attributo (sia in caso di ridondanza che di non ridondanza) avvengono sempre mediante l'accesso all'entità “Area Parcheggio”

Per risparmiare spazio si sceglie di **eliminare** l'attributo ridondante *#postiDisponibili*

# Traccia A

## Accessi con ridondanza

- Analizziamo gli accessi in caso di memorizzazione dell'attributo *#utentiAbilitati*

### Tavola degli accessi

	Concetto	Costrutto	Accesso	Tipo
OPI	Area Parcheggio	E	I	L
	Area Parcheggio	E	I	S
OP2	Contrattisti	SE	500	L
	Contrattisti	SE	2	S
	Utente	E	2	S
	Accesso	R	4	S
	Area Parcheggio	E	4	L
	Area Parcheggio	E	4	S
OP3	Area Parcheggio	E	10	L

$$\begin{aligned} \text{OP1): } 1L + 1S &= 3 \times 44.000 \\ &= 132.000/\text{mese} \end{aligned}$$

$$\begin{aligned} \text{OP2): } 504L + 12S &= 528 \times 22 \\ &= 11.616/\text{mese} \end{aligned}$$

$$\text{OP3): } 10L = 10 \times 1 = 10/\text{mese}$$

Totale: **143.626 accessi/mese**

# Traccia A

## Accessi senza ridondanza

- Analizziamo gli accessi in caso di cancellazione dell'attributo *#utentiAbilitati*

### Tavola degli accessi

	Concetto	Costrutto	Accesso	Tipo
OPI	Area Parcheggio	E	I	L
	Area Parcheggio	E	I	S
OP2	Contrattisti	SE	500	L
	Contrattisti	SE	2	S
	Utente	E	2	S
	Accesso	R	4	S
OP3	Area Parcheggio	E	10	L
	Accesso	R	8500	L

$$\text{OP1): } 1L + 1S = 3 \times 44.000 \\ = 132.000/\text{mese}$$

$$\text{OP2): } 500L + 8S = 516 \times 22 \\ = 11.352/\text{mese}$$

$$\text{OP3): } 6010L = 8510 \times 1 \\ = 8510/\text{mese}$$

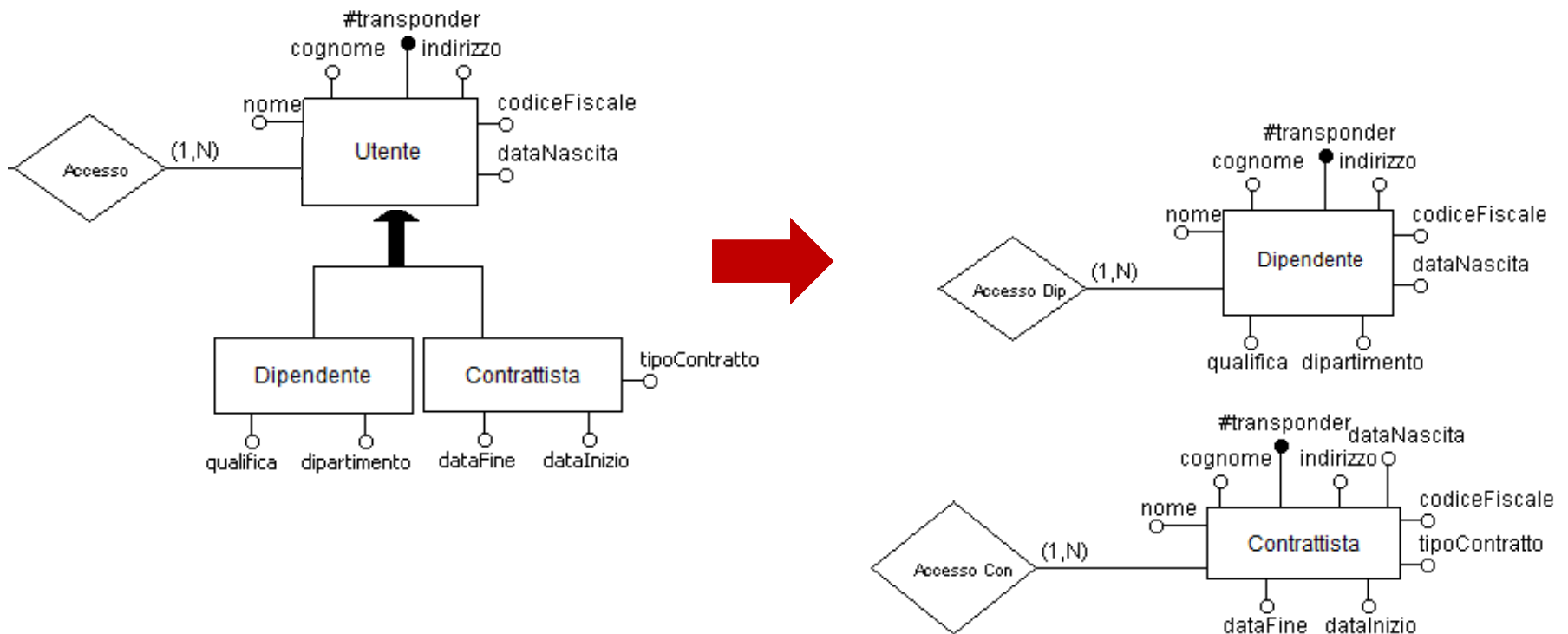
Totale: **151.862** accessi/mese

si sceglie di **mantenere** l'attributo ridondante *#utentiAbilitati*

# Traccia A

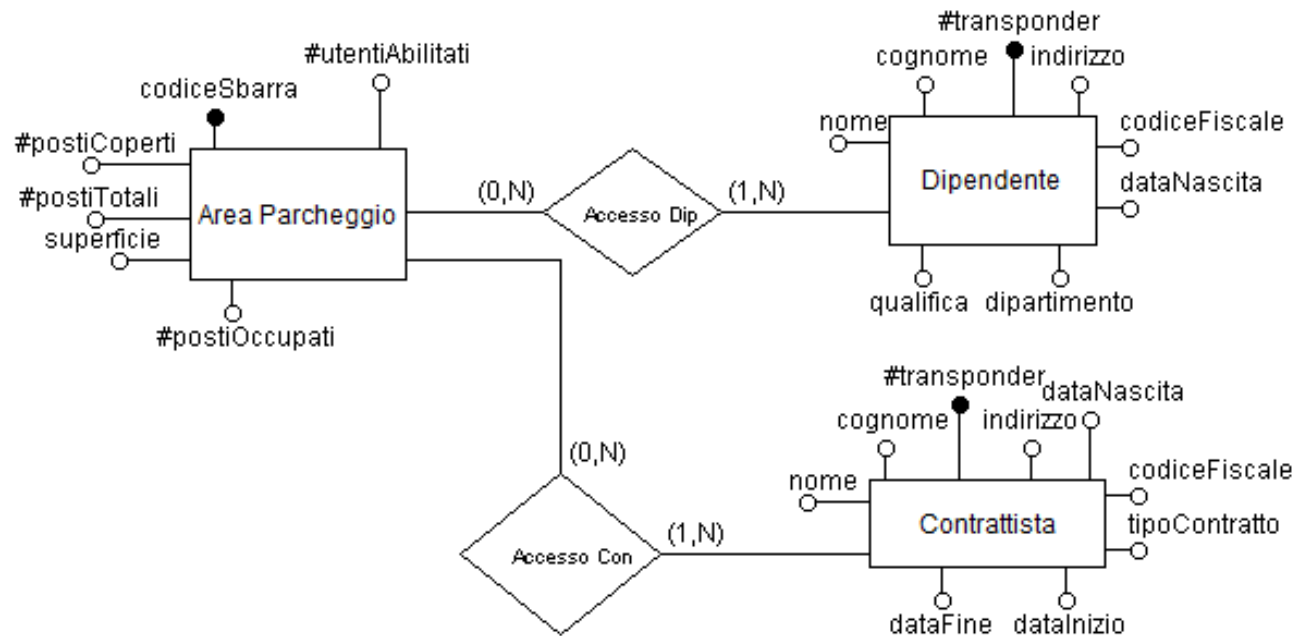
## Rimozione delle gerarchie

- Nello schema è presente un'unica gerarchia  
si sceglie di **accorpare** il padre nelle figlie



# Traccia A

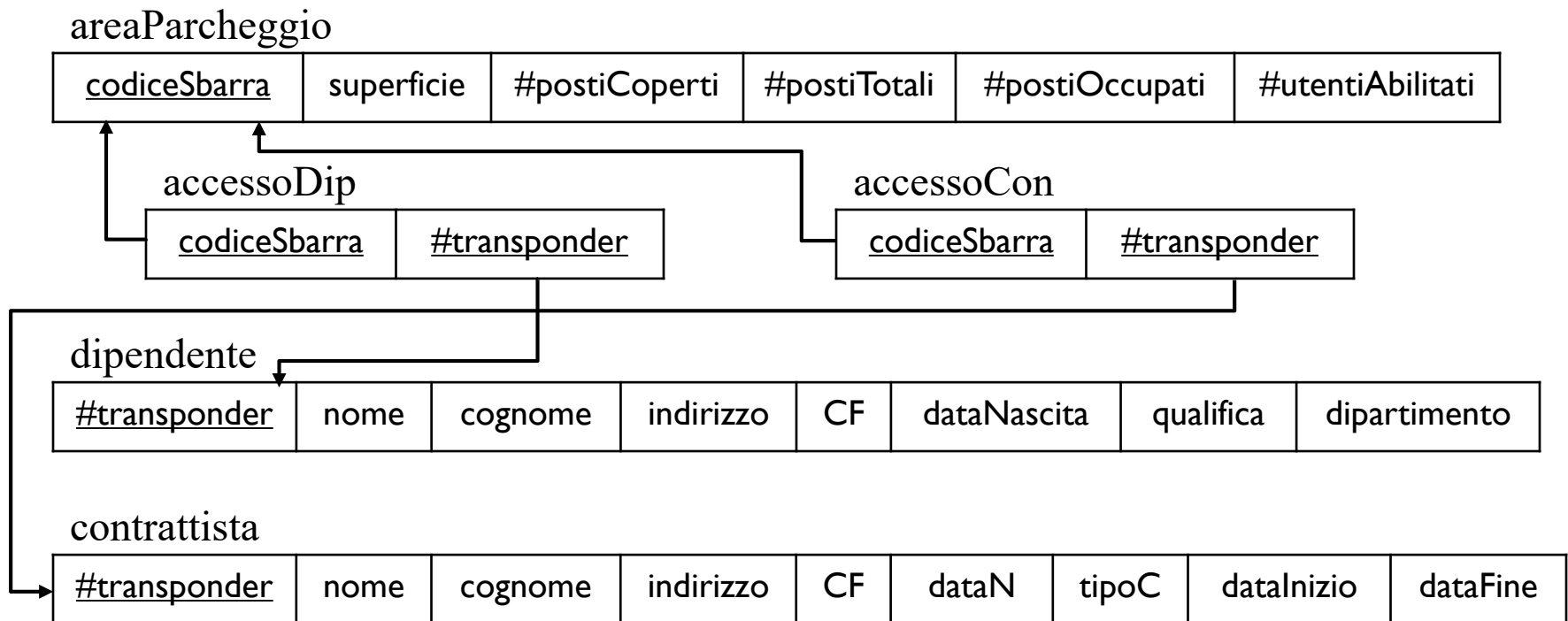
## Schema concettuale ristrutturato



# Traccia A

## Schema logico relazionale

Tradurre lo schema ristrutturato prodotto in uno schema logico relazionale, evitando tabelle ridondanti, e specificando vincoli, chiavi primarie e chiavi esterne.





# Traccia A

---

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

```
CREATE TABLE areaParcheggio(  
    codiceSbarra      CHAR(6) PRIMARY KEY,  
    superficie        INT,  
    #postiCoperti     INT,  
    #postiTotali      INT NOT NULL,  
    #postiOccupati    INT NOT NULL,  
    #utentiAbilitati  INT NOT NULL));
```

# Traccia A

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

**CREATE TABLE** dipendente(  
#transponder CHAR(6) PRIMARY KEY,  
nome VARCHAR(50) NOT NULL,  
cognome VARCHAR(50) NOT NULL,  
indirizzo VARCHAR(255),  
CF CHAR(16),  
dataNascita DATE,  
qualifica VARCHAR(50) NOT NULL,  
dipartimento VARCHAR(50) NOT NULL));

**CREATE TABLE** contrattista(  
#transponder CHAR(6) PRIMARY KEY,  
nome VARCHAR(50) NOT NULL,  
cognome VARCHAR(50) NOT NULL,  
indirizzo VARCHAR(255),  
CF CHAR(16),  
dataN DATE,  
tipoC VARCHAR(50),  
dataInizio DATE NOT NULL,  
dataFine DATE NOT NULL));

# Traccia A

---

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

```
CREATE TABLE accessoDip(  
    #transponder CHAR(6),  
    codiceSbarra CHAR(6),  
    PRIMARY KEY (#transponder, codiceSbarra),  
    FOREIGN KEY (#transponder) REFERENCES dipendente(#transponder)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE,  
    FOREIGN KEY (codiceSbarra) REFERENCES areaParcheggio(codiceSbarra)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE));
```

# Traccia A

---

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

```
CREATE TABLE accessoCon(  
    #transponder CHAR(6),  
    codiceSbarra CHAR(6),  
    PRIMARY KEY (#transponder, codiceSbarra),  
    FOREIGN KEY (#transponder) REFERENCES contrattista(#transponder)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE,  
    FOREIGN KEY (codiceSbarra) REFERENCES areaParcheggio(codiceSbarra)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE));
```

# Traccia B

---

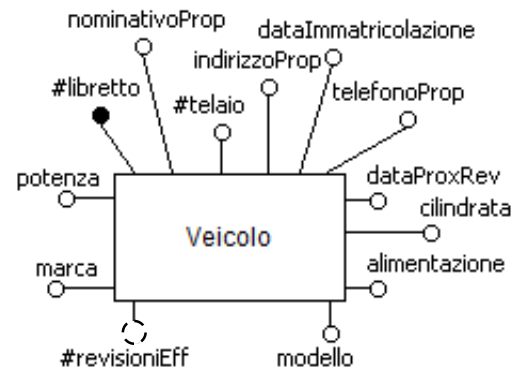
## Schema concettuale del database

Utilizzando il modello ER, progettare lo schema concettuale del database di un'officina abilitata alla revisione di veicoli. Esso dovrà memorizzare i dati dei veicoli revisionati e delle varie revisioni effettuate. Per ogni veicolo andranno memorizzati numero del libretto, numero di telaio, data di immatricolazione, dati anagrafici e di contatto del proprietario, marca, modello, alimentazione, cilindrata, potenza, data della prossima revisione ed il numero di revisioni effettuate presso l'officina. Inoltre, per gli autocarri andrà memorizzato il peso ed numero di assi, mentre per gli autoveicoli andrà memorizzata la categoria delle emissioni (Euro 4, Euro 5 o altri) ed il tipo di catalizzatore (con filtro antiparticolato autorigenerante, ad esaurimento, o altri tipi). Per gli altri tipi di veicoli non andrà memorizzato altro. Per ogni revisione effettuata andranno memorizzati data, costo e l'esito.

# Traccia B

## Schema concettuale del database

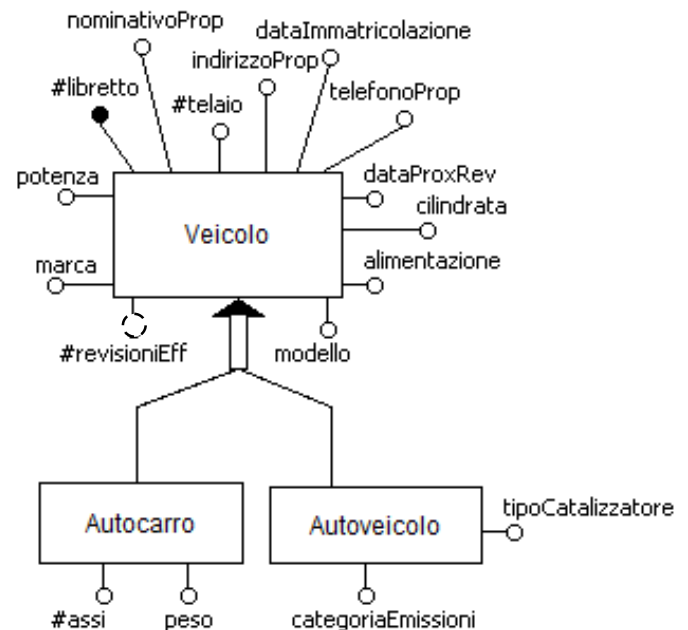
[...] Per ogni **veicolo** andranno memorizzati *numero del libretto*, *numero di telaio*, *data di immatricolazione*, *dati anagrafici e di contatto del proprietario*, *marca*, *modello*, *alimentazione*, *cilindrata*, *potenza*, *data della prossima revisione* ed il *numero di revisioni effettuate presso l'officina*. [...]



# Traccia B

## Schema concettuale del database

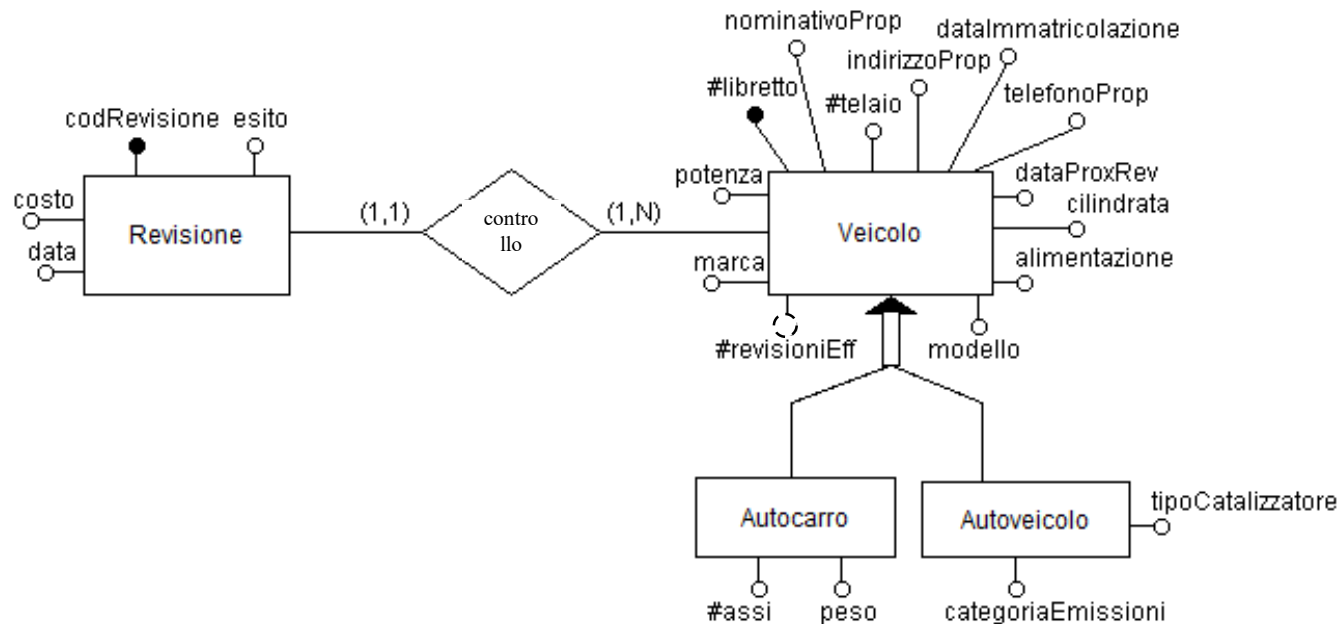
[...] per gli **autocarri** andrà memorizzato il *peso* ed *numero di assi*, mentre per gli autoveicoli andrà memorizzata la *categoria delle emissioni* (Euro 4, Euro 5 o altri) ed il *tipo di catalizzatore* (con filtro antiparticolato autorigenerante, ad esaurimento, o altri tipi). Per gli altri tipi di veicoli non andrà memorizzato altro.[...]



# Traccia B

## Schema concettuale del database

[...] Per ogni **revisione effettuata** andranno memorizzati *data*, *costo* e *l'esito*.





# Traccia B

---

## Ristrutturazione

Calcolare il carico applicativo e ristrutturare lo schema ER prodotto, valutando se mantenere l'attributo ridondante relativo al numero di revisioni effettuate. L'officina mantiene i dati dei veicoli e delle revisioni effettuate presso l'officina fino alla data di cancellazione dal PRA (Pubblico Registro Automobilistico). In particolare, il database memorizza i dati di circa 1.000 veicoli, di cui 800 sono autoveicoli e 100 autocarri. In media ciascun autocarro ha effettuato circa 4 revisioni presso l'officina, mentre i restanti veicoli ne hanno effettuate 2. Le tre (su 15) più frequenti sono:

OP1) Revisione veicolo (2 volte/giorno lavorativo). Una revisione su 10 riguarda un veicolo non memorizzato nel database. Per gli altri veicoli i dati tecnici vengono letti dal database.

OP2) Al termine di un giorno lavorativo, verifica scadenza della revisione dei veicoli memorizzati nel database, onde avvisare i proprietari dei veicoli con revisione in scadenza.

OP3) Stampa semestrale di un report che mostri i dati di ciascun veicolo, incluso il numero di revisioni effettuate presso l'officina.

# Traccia B

## Carico Applicativo

- ▶ il database memorizza i dati di circa 1.000 veicoli, di cui
  - ▶ 800 sono autoveicoli e
  - ▶ 100 autocarri,
- ▶ in media, ciascun autocarro ha effettuato circa 4 revisioni presso l'officina
- ▶ i restanti veicoli ne hanno effettuate 2.

**Tavola dei volumi**

Concetto	Costrutto	Volume
Veicolo	E	1000
Autocarro	SE	100
Autoveicolo	SE	800
Revisione	E	2200
Controllo	R	2200

$$(100 \times 4) + (900 \times 2)$$

# Traccia B

## Carico Applicativo

► Le principali operazioni da eseguire su questo database sono 15, di cui le tre più frequenti sono:

- OP1) Revisione veicolo (2 volte/giorno lavorativo). Una revisione su 10 riguarda un veicolo non memorizzato nel database. Per gli altri veicoli i dati tecnici vengono letti dal database.
- OP2) Al termine di un giorno lavorativo, verifica scadenza della revisione dei veicoli memorizzati nel database, onde avvisare i proprietari dei veicoli con revisione in scadenza.
- OP3) Stampa semestrale di un report che mostri i dati di ciascun veicolo, incluso il numero di revisioni effettuate presso l'officina.

### Tavola delle operazioni

Operazione	Tipo	Frequenza
OP1	I	2/giorno lavorativo
OP2	B	1/giorno lavorativo
OP3	B	1/semestre

$(2 \times 22) = 264/\text{semestre}$

132/semestre

# Traccia B

## Accessi con ridondanza

- Analizziamo gli accessi in caso di memorizzazione dell'attributo *#revisioniEff*

### Tavola degli accessi

	Concetto	Costrutto	Accesso	Tipo
OP1	Revisione	E	I	S
	Controllo	R	I	S
	Veicolo	E	I(*0,9)	L
	Veicolo	E	(0,I)+(0,9)=I	S
OP2	Veicolo	E	I000	L
OP3	Veicolo	E	I000	L

$$\text{OP1): } 1L(*0,9) + 3S = 6,9 \times 264 \approx 1822/\text{semestre}$$

$$\text{OP2): } 1000L = 1000 \times 132 = 132.000/\text{semestre}$$

$$\text{OP3): } 1000L = 1000 \times 1 = 1000/\text{semestre}$$

Totale: **134.822 accessi/semestre**

# Traccia B

## Accessi senza ridondanza

- Analizziamo gli accessi in caso di cancellazione dell'attributo *#revisioniEff*

Tavola degli accessi

	Concetto	Costrutto	Accesso	Tipo
OPI	Revisione	E	I	S
	Controllo	R	I	S
	Veicolo	E	I(*0,1)	S
	Veicolo	E	I(*0,9)	L
OP2	Veicolo	E	1000	L
OP3	Veicolo	E	1000	L
	Controllo	R	2200	L

$$\begin{aligned} \text{OP1): } & 2S + 1S(*0,1) + 1L(*0,9) \\ & = (4+0,2+0,9) \times 264 \\ & \approx 1346/\text{semestre} \end{aligned}$$

$$\begin{aligned} \text{OP2): } & 1000L = 1000 \times 132 \\ & = 132.000/\text{semestre} \end{aligned}$$

$$\begin{aligned} \text{OP3): } & 3200L = 3200 \times 1 \\ & = 3200/\text{semestre} \end{aligned}$$

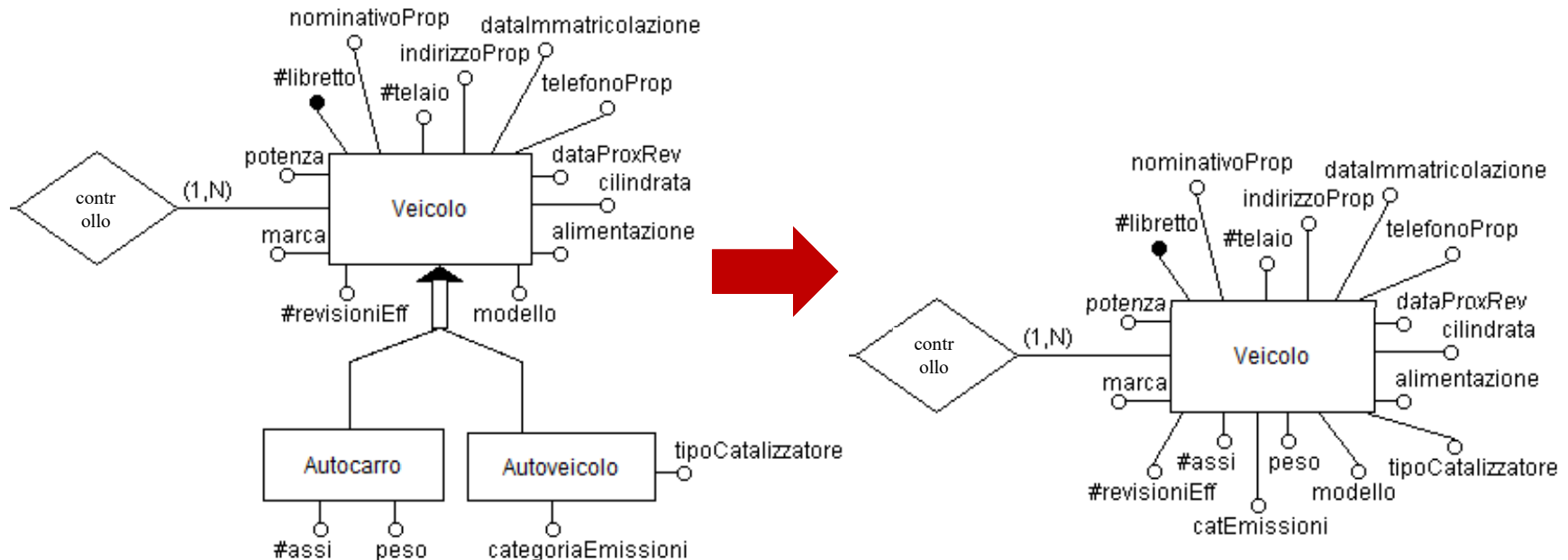
Totale: **136.546** accessi/semestre

si sceglie di **mantenere** l'attributo ridondante *#revisioniEff*

# Traccia B

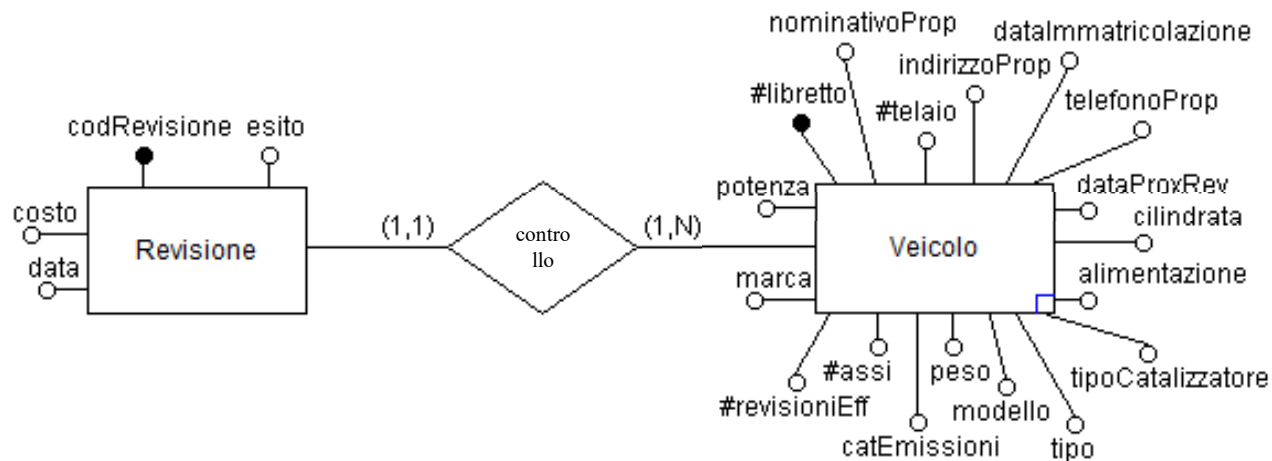
## Rimozione delle gerarchie

- Nello schema è presente un'unica gerarchia  
si sceglie di **accorpare** le figlie nel padre



# Traccia B

## Schema concettuale ristrutturato



# Traccia B

## Schema logico relazionale

Tradurre lo schema ristrutturato prodotto in uno schema logico relazionale, evitando tabelle ridondanti, e specificando vincoli, chiavi primarie e chiavi esterne.

revisione

<u>codRevisione</u>	data	costo	esito	#libretto
---------------------	------	-------	-------	-----------

veicolo

<u>#libretto</u>	nominativoProp	indirizzoProp	telefonoProp	dataImmatr	#telaio	
	potenza	#revisioniEff	modello	alimentazione	marca	tipo
	cilindrata	dataProxRev	catEmissioni	tipoCatalizzat	#assi	peso



# Traccia A

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

```
CREATE TABLE veicolo(  
    #libretto      CHAR(15) PRIMARY KEY,  
    nominativoProp VARCHAR(100) NOT NULL,    ...  
    indirizzoProp  VARCHAR(100),              alimentazione VARCHAR(20) NOT NULL,  
    telefonoProp   VARCHAR(11),                marca             VARCHAR(20) NOT NULL,  
    #telaio        CHAR(6),                    cilindrata        INT NOT NULL,  
    dataImmatr     DATE NOT NULL,              dataProxRev       DATE,  
    potenza        INT NOT NULL,               dataN             DATE,  
    #revisioniEff  INT NOT NULL,               catEmissioni      VARCHAR(10),  
    modello        VARCHAR(30) NOT NULL,        tipoCataliz       VARCHAR(20),  
    ...           ...                           #assi             INT,  
    ...           ...                           peso              INT));
```

# Traccia B

---

## Creazione del DB mediante DDL di SQL

Tradurre nel DDL di SQL lo schema logico prodotto

```
CREATE TABLE revisione(  
    codRevisione    CHAR(6) PRIMARY KEY,  
    data            DATE NOT NULL,  
    costo           NUMERIC(7,2) NOT NULL,  
    esito           VARCHAR(255),  
    #libretto       CHAR(15) NOT NULL,  
    FOREIGN KEY (#libretto) REFERENCES veicolo(#libretto)  
                                ON UPDATE CASCADE  
                                ON DELETE CASCADE));
```