

*Basi di dati A.A. 2019/2020*  
*Progettazione Base di Dati:*  
***Car rental***

Callegari Sara

Molon Eleonora

Trabucco Giovanni

# Indice

<b>1 DESCRIZIONE DEL PROGETTO</b>	<b>2</b>
1.1 Requisiti strutturali	2
1.2 Operazione sulla base di dati	3
1.3 Glossario	4
<b>2 PROGETTAZIONE CONCETTUALE</b>	<b>5</b>
2.1 Modello concettuale: Entità-Associazione (E-R)	5
2.2 Dizionario dei dati	6
2.2.1 Entità	6
2.2.2 Associazioni	6
2.3 Regole di vincolo	7
<b>3 PROGETTAZIONE LOGICA</b>	<b>7</b>
3.1 Ristrutturazione schema ER	7
3.2 Modello logico: Relazionale	7
3.3 Schema logico: Regole di vincolo	8
<b>4 CODICE SQL</b>	<b>8</b>
4.1 Struttura	8
4.2 Query	10

# 1 DESCRIZIONE DEL PROGETTO

Si vuole realizzare una base di dati per un'agenzia che noleggia auto di vari modelli e che possiede varie sedi in tutta Italia. Si vuole tenere traccia di tutti i dettagli dei contratti di noleggio stipulati dall'azienda in questione.

## 1.1 Requisiti strutturali

### **Fraasi per Dipendente:**

Per ogni dipendente, identificato in maniera univoca dal codice fiscale, rappresentiamo il nome, il cognome e lo stipendio.

Un dipendente lavora in una sola sede: non vogliamo tenere lo storico dei dipendenti che hanno lavorato in un centro.

### **Fraasi per Centro:**

Per ogni centro, identificato univocamente da un codice ID, rappresentiamo la città in cui si trova e il relativo indirizzo.

In un centro lavorano più dipendenti e si trovano diverse auto. I centri vengono utilizzati come luogo di partenza e luogo di arrivo di un'auto.

### **Fraasi per Contratto:**

Per ogni contratto, identificato in modo univoco dal numero della fattura emessa, rappresentiamo il tipo di assicurazione collegata, che può essere:

- base: se comprende solo la RCA
- parziale: se comprende la RCA e un'altra assicurazione a scelta
- totale: se comprende, oltre alla RCA, anche tutte le altre assicurazioni

L'assicurazione base è compresa nel prezzo, mentre se viene scelta una tipologia di assicurazione diversa si ha un sovrapprezzo sul prezzo del noleggio: +100 euro per la parziale e +170 per la totale.

Un contratto è stipulato con un solo cliente ed è relativo a un solo veicolo. Il contratto stabilisce che il centro di partenza, luogo in cui si ritira l'auto, possa essere diverso dal centro di arrivo, dove si restituisce l'auto al termine del noleggio.

### **Fraasi per Cliente:**

Ogni cliente viene rappresentato univocamente dal numero di patente. Inoltre, si rappresentano altri dati personali: il codice fiscale, che dev'essere univoco per ciascun cliente, il nome, il cognome e il numero di telefono, anch'esso unico.

Un cliente può firmare diversi contratti per la stessa auto o per auto diverse. Un cliente, quando riconsegna un'automobile presa a noleggio, può recensire quella determinata auto. Dopo al massimo qualche giorno, la recensione viene inserita nella base di dati.

### **Fraasi per Auto:**

Per ogni auto, identificata in maniera univoca dalla targa, si rappresentano la marca, il modello, l'anno di immatricolazione, il numero di posti che possiede, il prezzo giornaliero del noleggio e lo stato del veicolo, che può essere:

- libero: se il veicolo è disponibile per il noleggio
- occupato: se il veicolo è momentaneamente già noleggiato

- in arrivo: se il veicolo è stato acquistato ma non è ancora disponibile
- in manutenzione: se il veicolo non è momentaneamente disponibile causa riparazione

In ogni momento, un'auto si trova al massimo in un centro. Ogni auto viene noleggiata più volte. Inoltre, l'azienda sceglie di inserire nel database automobili che ha già acquistato, ma che non ha ancora a disposizione. Ogni auto appartiene a una categoria di veicoli. Inoltre, ogni auto dispone di vari servizi.

### **Fraasi per Categoria:**

Per ogni categoria, rappresentata da una sigla univoca, si rappresenta una descrizione.

Una categoria comprende vari veicoli. Inoltre, l'azienda decide di inserire nel database nuove categorie che comprenderanno auto future che non sono state ancora acquistare.

### **Fraasi per Accessorio:**

Per ogni accessorio, identificato univocamente da una sigla, si rappresenta una descrizione.

Più auto possono disporre di un determinato accessorio.

### **Fraasi per Recensione:**

Per ogni recensione lasciata da un cliente per una determinata auto, si registra il voto e l'eventuale opinione lasciata. Inoltre, si rappresenta la data in cui la recensione viene lasciata.

## **1.2 Operazione sulla base di dati**

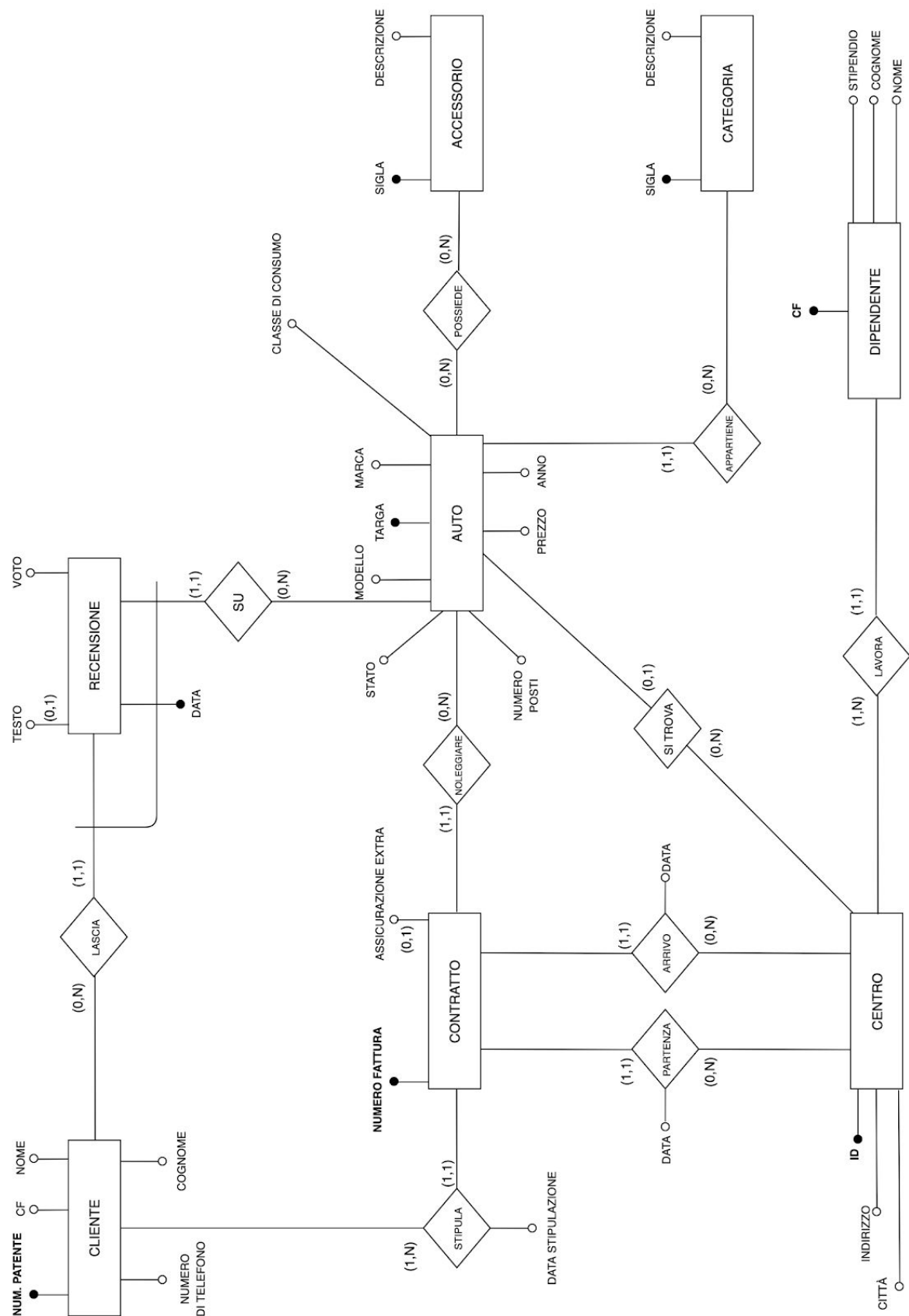
<b>Operazione</b>	<b>Tipologia</b>	<b>Frequenza</b>
1) Trovare il centro che spende di più per gli stipendi dei dipendenti	Interrogazione	1/Mese
2) Elencare le auto libere in un centro che appartengono a una data categoria	Interrogazione	1/Giorno
3) Mostrare l'accessorio più presente nelle auto	Interrogazione	4/Anno
4) Contratto con il ricavo più alto	Interrogazione	1/Anno
5) Per ogni auto, elenco delle recensioni ordinate per ordine di voto medio decrescente	Interrogazione	2/Settimana
6) Città dalla quale partono il maggior numero di auto ibride	Interrogazione	1/Mese
7) Centro con più traffico di auto	Interrogazione	1/Anno
Aggiornamento auto	Modifica	4/Anno
Aggiornamento contratti	Modifica	1/Giorno
Aggiornamento accessori	Modifica	4/Anno
Aggiornamento categorie	Modifica	4/Anno
Aggiornamento recensioni	Modifica	1/Settimana
Aggiornamento centri	Modifica	1/Anno

### 1.3 Glossario

Termine	Descrizione	Collegamenti
Cliente	Chi noleggia un' auto	Contratto, Recensione
Contratto	Documento che sancisce e registra l'atto nel noleggio	Cliente, Centro, Auto
Auto	Veicolo noleggiabile da un cliente	Contratto, Centro, Accessorio, Categoria, Recensione
Accessorio	Oggetto che completa e/o arricchisce le funzionalità di un' auto	Auto
Categoria	Insieme di auto con caratteristiche simili	Auto
Centro	Stabilimento fisico dove le auto vengono noleggate e riconsegnate	Dipendente, Contratto, Auto
Recensione	Opinione che un cliente lascia su un'auto	Cliente, Auto
Dipendente	Chi lavora in un centro	Centro

# 2 PROGETTAZIONE CONCETTUALE

## 2.1 Modello concettuale: Entità-Associazione (E-R)



## 2.2 Dizionario dei dati

### 2.2.1 Entità

Entità	Descrizione	Attributi	Identificatore
CLIENTE	Noleggia un' auto	Numero patente, CF, Nome, Cognome, Numero di Telefono	Numero Patente
CONTRATTO	Sancisce l'avvenuto noleggio di un' auto	Numero fattura, Assicurazione Extra	Numero Fattura
AUTO	Viene noleggiata	Targa, Prezzo, Marca, Modello, Anno, Stato, Numero posti	Targa
ACCESSORIO	Può essere presente o meno in un'auto	Sigla, Descrizione	Sigla
CATEGORIA	Insieme di auto con caratteristiche simili	Sigla, Descrizione	Sigla
CENTRO	Dove le auto vengono noleggiate e restituite	ID, Città, Indirizzo	ID
DIPENDENTE	Lavora per la nostra azienda	CF, Nome, Cognome, Stipendio	CF
RECENSIONE	Opinione che un cliente lascia su un'auto	Data, voto, testo	Cliente, Auto, Data

### 2.2.2 Associazioni

Associazioni	Entità collegate
Stipula	Cliente (1,N), Contratto (1,1)
Lascia	Cliente (0,N), Recensione(1,1)
Noleggiare	Contratto (1,1), Auto (0,N)
Partenza	Contratto (1,1), Centro (0,N)
Arrivo	Contratto (1,1), Centro (0,N)
Si trova	Auto (0,1), Centro (0,N)
Appartiene	Auto (1,1), Categoria (0,N)
Possiede	Auto (0,N), Accessorio (0,N)
Lavora	Dipendente (1,1), Centro (1,N)
Su	Recensione (1,1), Auto (0,N)

## 2.3 Regole di vincolo

- Non sono presenti accessori con la stessa sigla
- Non sono presenti categorie con la stessa sigla
- Non sono presenti centri con lo stesso ID
- Il numero della fattura è un progressivo, pertanto non vi sono contratti con lo stesso identificativo

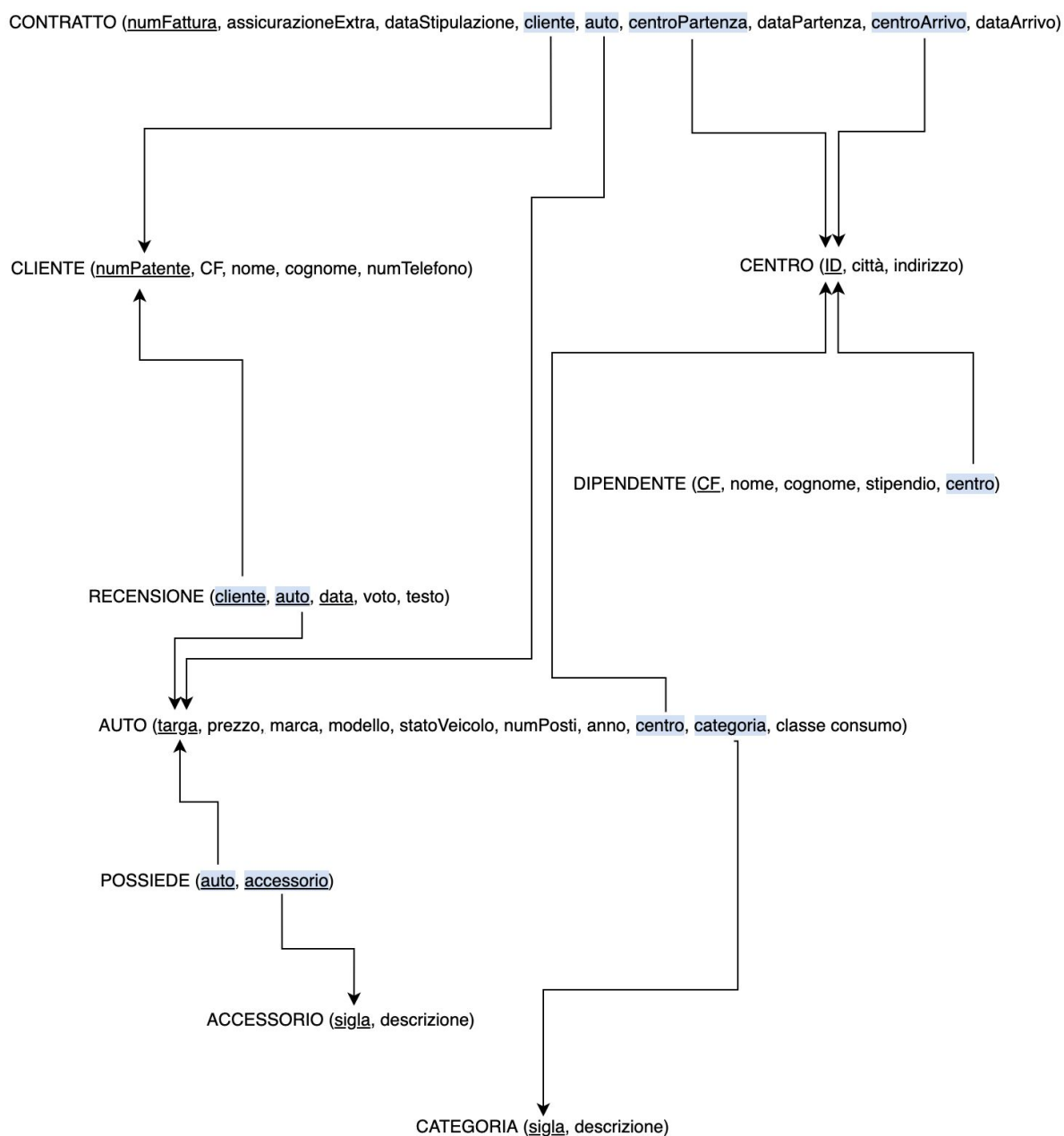
## 3 PROGETTAZIONE LOGICA

### 3.1 Ristrutturazione schema ER

Per come abbiamo costruito lo schema ER, non riteniamo sia necessario effettuare alcun tipo di ristrutturazione.

### 3.2 Modello logico: Relazionale

Di seguito lo schema logico prodotto per la rappresentazione della realtà di interesse:





### 3.3 Schema logico: Regole di vincolo

- Gli attributi di Contratto non devono essere nulli
- Gli attributi di Cliente non devono essere nulli, CF e numero di telefono sono unici
- Gli attributi di Centro non devono essere nulli
- Gli attributi di Dipendente non devono essere nulli
- Gli attributi di Recensione, ad eccezione di 'testo', non devono essere nulli
- Gli attributi di Auto non devono essere nulli, ad eccezione di 'centro'
- Gli attributi di Possiede non devono essere nulli
- Gli attributi di Accessorio non devono essere nulli
- Gli attributi di Categoria non devono essere nulli

## 4 CODICE SQL

### 4.1 Struttura

```
CREATE TABLE centro
(
    id character (3) NOT NULL,
    citta character varying (20) NOT NULL,
    indirizzo character varying (50) NOT NULL,
    CONSTRAINT "centro_PK" PRIMARY KEY ( id )
);

CREATE TABLE categoria
(
    sigla character varying (20) NOT NULL,
    descrizione character varying (150) NOT NULL,
    CONSTRAINT "categoria_PK" PRIMARY KEY ( sigla )
);

CREATE TABLE accessorio
(
    sigla character (2) NOT NULL,
    descrizione character varying (250) NOT NULL,
    CONSTRAINT "accessorio_PK" PRIMARY KEY ( sigla )
);

CREATE TABLE cliente
(
    NumPatente character (10) NOT NULL,
    CF character (16) NOT NULL UNIQUE,
    nome character varying (20) NOT NULL,
    cognome character varying (20) NOT NULL,
    numTelefono numeric (10) NOT NULL UNIQUE,
    CONSTRAINT "cliente_PK" PRIMARY KEY ( numPatente )
);

CREATE TABLE dipendente
(
    CF character (16) NOT NULL ,
    nome character varying (20) NOT NULL ,
    cognome character varying (20) NOT NULL ,
    stipendio numeric (6,2) NOT NULL ,
    centro character (3) NOT NULL,
    CONSTRAINT "dipendente_PK" PRIMARY KEY (CF) ,
    CONSTRAINT "dipendente_centro_FK" FOREIGN KEY (centro)
        REFERENCES centro (ID) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT
);
```

```

CREATE TABLE auto
(
    targa character (7) NOT NULL,
    prezzo numeric (3) NOT NULL,
    marca character varying (20) NOT NULL,
    modello character varying (20) NOT NULL,
    statoVeicolo character varying (20) NOT NULL CHECK (statoVeicolo='libero' OR
    statoVeicolo='occupato' OR statoVeicolo='in arrivo' OR statoVeicolo='in manutenzione'),
    numPosti numeric (2) NOT NULL,
    anno numeric (4) NOT NULL,
    centro character (3),
    categoria character varying (20) NOT NULL,
    classeConsumo character varying (10) NOT NULL
    CONSTRAINT "auto_PK" PRIMARY KEY ( targa ),
    CONSTRAINT "auto_centro_FK" FOREIGN KEY ( centro )
        REFERENCES centro ( ID ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "auto_categoria_FK" FOREIGN KEY ( categoria )
        REFERENCES categoria ( sigla ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT
);

CREATE TABLE possiede
(
    auto character (7) NOT NULL,
    accessorio character (2) NOT NULL,
    CONSTRAINT "possiede_PK" PRIMARY KEY ( auto, accessorio ),
    CONSTRAINT "possiede_accessorio_FK" FOREIGN KEY ( accessorio )
        REFERENCES accessorio(sigla) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "possiede_auto_FK" FOREIGN KEY ( auto )
        REFERENCES auto( targa ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT
);

CREATE TABLE recensione
(
    cliente character (10) NOT NULL ,
    auto character (7) NOT NULL,
    voto numeric (1) NOT NULL,
    testo character varying (500),
    data DATE NOT NULL,
    CONSTRAINT "recensisce_PK" PRIMARY KEY ( cliente, auto, data),
    CONSTRAINT "recensisce_auto_FK" FOREIGN KEY ( auto )
        REFERENCES auto( targa ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "recensisce_cliente_FK" FOREIGN KEY ( cliente )
        REFERENCES cliente( NumPatente ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT
);

```

```

CREATE TABLE contratto
(
    numFattura numeric (4) NOT NULL,
    assicurazioneExtra character varying (15) CHECK (assicurazioneExtra='Base'
OR assicurazioneExtra='Parziale' OR assicurazioneExtra='Totale'),
    dataStipulazione date NOT NULL,
    cliente character (16) NOT NULL,
    auto character (7) NOT NULL,
    centroPartenza character (3) NOT NULL,
    dataPartenza date NOT NULL,
    centroArrivo character (3) NOT NULL,
    dataArrivo date NOT NULL,
    CONSTRAINT "contratto_PK" PRIMARY KEY ( numFattura ),
    CONSTRAINT "contratto_cliente" FOREIGN KEY ( cliente )
        REFERENCES cliente ( numPatente ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "contratto_auto_FK" FOREIGN KEY ( auto )
        REFERENCES auto ( targa ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "contratto_centroPartenza_FK" FOREIGN KEY ( centroPartenza )
        REFERENCES centro ( ID ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT,
    CONSTRAINT "contratto_centroArrivo_FK" FOREIGN KEY ( centroArrivo )
        REFERENCES centro ( ID ) MATCH SIMPLE
        ON UPDATE CASCADE ON DELETE RESTRICT
);

```

## 4.2 Query

1. Mostrare il centro che spende mensilmente di più per gli stipendi dei suoi dipendenti e la somma totale

```

CREATE VIEW stipendi AS(
    SELECT centro, SUM(stipendio) AS costo_stipendi
    FROM dipendente
    GROUP BY centro
);
SELECT *
FROM stipendi
WHERE costo_stipendi = (
    SELECT MAX(costo_stipendi)
    FROM stipendi
);

```

2. Dati un centro 'x' e una categoria 'y': elenco delle auto libere

```

SELECT *
FROM auto
WHERE categoria='$categoria'
AND statoveicolo='libero'
AND centro='$centro';

```

3. Mostrare l'accessorio più richiesto e in quante auto compare

```

SELECT accessorio, COUNT(auto) AS numero_auto
FROM possiede
GROUP BY accessorio
ORDER BY numero_auto DESC
LIMIT 1;

```



4. Mostrare il contratto con il ricavo totale più elevato e l'auto noleggiata

```
CREATE VIEW costo AS (  
    SELECT numfattura, (dataarrivo-datapartenza)*prezzo AS fatturato, auto  
    FROM contratto JOIN auto ON contratto.auto = auto.targa  
);  
  
SELECT *  
FROM costo  
WHERE fatturato = (  
    SELECT MAX(fatturato)  
    FROM costo  
);
```

5. Per ogni auto mostrare in ordine decrescente il voto medio ottenuto nelle recensioni

```
SELECT auto, AVG(voto) AS media  
FROM recensione  
WHERE data >= '2019-01-01' AND data <= '2019-12-31'  
GROUP BY auto  
ORDER BY media DESC;
```

6. In quale città sono state prese più auto ibride

```
SELECT citta, COUNT(auto) AS numero_auto_ibride  
FROM centro JOIN contratto ON centro.ID = contratto.centropartenza  
JOIN auto ON contratto.auto = auto.targa  
WHERE auto.categoria = 'Ibrida'  
GROUP BY citta  
ORDER BY numero_auto_ibride DESC;
```

7. Il centro con più traffico (cioè con più partenze e più arrivi)

```
CREATE VIEW conta_part AS (  
    SELECT centropartenza AS centro, count(*) AS traffico_part  
    FROM contratto  
    GROUP BY centropartenza);  
  
CREATE VIEW conta_arr AS (  
    SELECT centroarrivo AS centro, count(*) AS traffico_arr  
    FROM contratto  
    GROUP BY centroarrivo);  
  
SELECT centro, traffico_part+traffico_arr AS traffico_totale  
FROM conta_part NATURAL JOIN conta_arr  
WHERE (traffico_part+traffico_arr) = (  
    SELECT MAX (traffico_part+traffico_arr)  
    FROM conta_part NATURAL JOIN conta_arr  
);
```