

Con riferimento al DB in figura:

- Creare in SQL la tabella MEDICI definendo la chiave primaria e le chiavi esterne
- Ottenere in SQL Cognome e Nome dei pazienti nati a Padova prima del 1979, in ordine alfabetico
- Ottenere in SQL Cognome e Telefono del primario del reparto BETA
- Ottenere in SQL il numero di operazioni che sono state effettuate in Veneto a tariffa >1500

PAZIENTI

cod_paz	cognome	nome	data_nascita	città
1001	Preti	Maurizio	11/04/85	Milano
1002	Giacomini	Francesco	11/03/72	Vicenza
1003	Hubner	Franco	23/12/66	Pordenone
1004	Gustavi	Gustavo	23/01/65	Padova
1005	Frati	Dino	01/03/46	Vicenza
1006	Sona	Alberto	01/05/71	Rovigo
1007	Anastasi	Maria	03/05/65	Udine
1008	Bettini	Giovanna	04/07/76	Padova
1009	Bentisa	Samuela	07/08/75	Padova
1010	Benini	Alessandro	07/01/80	Padova
1011	Novi	Giaco	05/12/67	Pordenone
1012	Bisco	Fabiana	03/03/56	Milano
1013	Denti	Roberta	04/07/73	Udine
1014	Costantini	Federico	03/02/86	Vicenza
1015	Laurenti	Guido	04/04/56	Rovigo

MEDICI

cod_medico	cognome	nome	cod_reparto	telefono
1	Pitteri	Mario	101	2356
2	Rossi	Carlo	101	2345
3	Bianchi	Giovanni	102	2341
4	Franchi	Gilda	102	2113
5	Magnolia	Francesca	103	2010
6	Zambarbieri	Eva	104	3212
7	Donati	Mauro	104	2189
8	Marangoni	Matteo	104	2189
9	Quasimodo	Salvatore	103	2010
10	Bossarlolo	Guido	103	2011

INTERVENTI

cod_intervento	descrizione	durata_standard	frequenza_standard	tariffa
52	RIPARAZIONE DI CHEILOSCHISI E DI PALATOSCHISI	5	0,3	2500
54	INTERVENTI SU SENI PARANASALI E MASTOIDE	4	0,2	2200
60	TONSILLECTOMIA	3	0,5	1000

REPARTI

cod_reparto	nome_reparto	cod_primario
101	ALFA	2
102	BETA	4
103	GAMMA	9
104	DELTA	7

GEOGRAFIA

Città	regione
Udine	Friuli
Pordenone	Friuli
Milano	Lombardia
Vicenza	Veneto
Rovigo	Veneto
Padova	Veneto

OPERAZIONI

cod_paz	cod_medico	data_ingresso	cod_intervento	durata_ricovero
1001	2	01/04/05	52	6
1002	4	01/03/05	52	4
1003	9	04/03/05	52	5
1004	7	07/07/05	52	8
1005	7	06/06/05	52	3
1015	3	01/08/05	52	6
1012	1	10/10/05	52	4
1011	5	03/05/05	52	4
1006	3	01/08/05	54	3
1007	2	01/07/05	54	5
1008	3	01/08/05	54	6
1014	2	05/08/05	54	5
1013	3	04/09/05	54	4
1009	1	23/02/05	60	2
1010	1	24/02/05	60	2
1001	2	27/02/05	60	3
1002	6	28/09/05	60	6
1003	5	26/05/05	60	4
1004	5	28/09/05	60	5
1015	2	01/02/05	60	3
1014	2	01/04/05	60	2
1013	9	07/05/05	60	3
1012	2	03/04/05	60	3
1005	2	03/02/05	60	3
1007	2	03/05/05	60	2

SOLUZIONI:

a)

```
CREATE TABLE medici_bis
(cod_med numeric,
cognome char(30),
nome char(30),
cod_rep integer,
telefono char(30),
PRIMARY KEY (cod_med),
FOREIGN KEY (cod_rep) REFERENCES reparti (cod_reparto)
);
```

b)

```
SELECT cognome, nome
FROM pazienti
WHERE data_nascita < #1/1/1979# AND città="padova"
ORDER BY cognome;
```

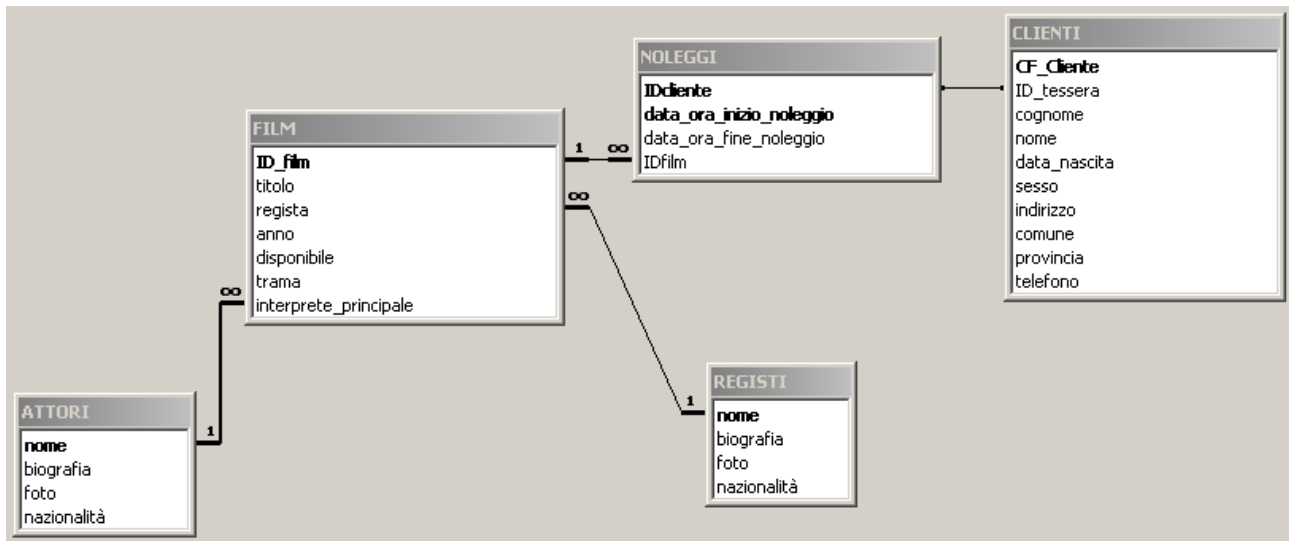
c)

```
SELECT cognome, telefono
FROM medici INNER JOIN reparti ON medici.cod_medico=reparti.cod_primario
WHERE reparti.nome_reparto="BETA";
```

d)

```
SELECT COUNT(*)
FROM (operazioni INNER JOIN interventi ON operazioni.cod_intervento=interventi.cod_intervento)
INNER JOIN pazienti ON pazienti.cod_paz=operazioni.cod_paz
WHERE città IN (SELECT città FROM geografia WHERE regione="Veneto") AND tariffa>1500;
```

Si consideri un DB contenente dati sui film presenti in una videoteca, le cui relazioni e associazioni sono individuate dalla figura seguente.



1. Si realizzi una Query che restituisca l'elenco in ordine alfabetico di tutti gli attori (senza ripetizioni).
2. Si realizzi una Query che restituisca, per ciascun regista, il nome del regista e il numero di film diretti.
3. Si realizzi una Query che restituisca il titolo dell'ultimo film noleggiato.
4. Si realizzi una Query che restituisca il numero di noleggi effettuati da clienti di sesso femminile e da clienti di sesso maschile.
5. Si realizzi una Query che restituisca il titolo, la trama, il regista, l'interprete principale, la biografia del regista e dell'interprete principale dei film diretti da registi che nella loro carriera sono stati anche interpreti principali di almeno un film. Si provi a realizzare la Query in 2 modi: con un inner join annidato e senza usare inner join.

SOLUZIONI

1.
SELECT DISTINCT nome
FROM ATTORI
ORDER BY nome;

NOTARE CHE DISTINCT NON E' INDISPENSABILE ESSENDO NOME LA CHIAVE PRIMARIA DELLA TABELLA

2.
SELECT regista, COUNT(*) AS numero_occorrenze
FROM FILM
GROUP BY regista;

3.
SELECT FILM.titolo
FROM NOLEGGI INNER JOIN FILM ON NOLEGGI.IDfilm=FILM.ID_film
WHERE (NOLEGGI.data_ora_inizio_noleggio)>=All (SELECT data_ora_inizio_noleggio FROM NOLEGGI)

4.
SELECT sesso, COUNT(*) AS numero_noleggi
FROM CLIENTI INNER JOIN NOLEGGI ON CLIENTI.ID_tessera=NOLEGGI.IDcliente
GROUP BY sesso;

5.

Sol. senza INNER JOIN

```
SELECT film.titolo, film.anno, film.trama, film.regista, registi.biografia, film.interprete_principale, attori.biografia
FROM registi, film, attori, film AS tab1
WHERE film.regista=registi.nome AND tab1.interprete_principale=registi.nome AND film.interprete_principale=attori.nome;
```

Sol. con INNER JOIN annidato

```
SELECT film.titolo, film.anno, film.trama, film.regista, registi.biografia, film.interprete_principale, attori.biografia
FROM ((registi INNER JOIN film AS tab1 ON tab1.interprete_principale=registi.nome) INNER JOIN film ON
film.regista=registi.nome) INNER JOIN attori ON film.interprete_principale=attori.nome;
```