

LABORATORIO DI BASI DI DATI: ESERCITAZIONE 3

ESERCIZIO 1

Con riferimento al database realizzato in PostgreSQL nel laboratorio precedente, relativo allo schema relazionale:

- *persona* (*id_persona*, *nome*, *cognome*)
- *film* (*id_film*, *id_regista*, *titolo*, *genere*, *anno*): dove *id_regista* è una chiave esterna che fa riferimento a *persona*;
- *partecipazione* (*id_attore*, *id_film*, *ruolo*): dove *id_attore* ed *id_film* sono chiavi esterne che fanno riferimento rispettivamente a *persona* e *film*;
- *cinema* (*id_cinema*, *nome*, *indirizzo*)
- *proiezione* (*id_cinema*, *id_film*, *giorno*): dove *id_cinema* e *id_film* sono chiavi esterne che fanno riferimento rispettivamente a *cinema* e *film*.

si scrivano in SQL le seguenti interrogazioni

- (1) Qual'è il contenuto della tabella *persona*?

Soluzione:

```
SELECT * FROM persona;
```

- (2) Selezionare i cognomi delle persone nella BD, senza eliminare i doppioni.

Soluzione:

```
SELECT cognome FROM persona;
```

- (3) Selezionare i cognomi delle persone ordinati in ordine alfabetico, senza eliminare i doppioni.

Soluzione:

```
SELECT cognome FROM persona ORDER BY cognome;
```

- (4) Definire l'insieme dei cognomi delle persone mantenute nella BD.

Soluzione:

```
SELECT DISTINCT cognome FROM persona;
```

- (5) Selezionare le persone di nome John mantenute nella BD.

Soluzione:

```
SELECT * FROM persona WHERE nome = 'John';
```

- (6) Selezionare i cognomi delle persone di nome John, mantenute nella BD.

Soluzione:

```
SELECT DISTINCT cognome FROM persona WHERE nome = 'John';
```

- (7) Selezionare le persone mantenute nella BD che sono attori.

Soluzione:

```
SELECT DISTINCT persona.* FROM persona, partecipazione  
WHERE persona.id_persona = partecipazione.id_attore;
```

- (8) Definire la lista di tutte le interpretazioni, precisando il nome dell'attore, il cognome dell'attore, il ruolo interpretato ed il titolo del film. Il risultato dovrà essere nella forma:

cognome	nome	ruolo	titolo
Nicole	Kidman	Grace	Dogville
Paul	Bettany	Tom Edison	Dogville
...

Soluzione:

```
SELECT p.cognome, p.nome, partecipazione.ruolo, film.titolo
FROM persona AS p, partecipazione, film
WHERE p.id_persona = partecipazione.id_attore
AND partecipazione.id_film = film.id_film;
```

- (9) Definire la lista di tutte le interpretazioni, precisando il nome dell'attore, il cognome dell'attore, il ruolo interpretato ed il titolo del film. Il risultato dovrà essere nella forma:

Nicole Kidman ha interpretato il ruolo di Grace nel film Dogville
 Paul Bettany ha interpretato il ruolo di Tom Edison nel film Dogville
 ...

Soluzione:

```
SELECT p.cognome || ' ' || p.nome || ' ha svolto il ruolo di '
|| pt.ruolo || ' nel film ' || f.titolo
FROM persona AS p, partecipazione AS pt, film AS f
WHERE p.id_persona = pt.id_attore
AND partecipazione.id_film = film.id_film;
```

- (10) Quali sono i titoli dei film di genere drammatico?

Soluzione:

```
SELECT DISTINCT titolo FROM film
WHERE genere = 'drammatico';
```

- (11) Quali film (titoli) sono stati proiettati nel 2002?

Soluzione:

```
SELECT DISTINCT film.titolo
FROM film NATURAL JOIN proiezione
WHERE proiezione.giorno >= '01/01/2002' AND
proiezione.giorno <= '31/12/2002';
```

- (12) Elencare i titoli dei film diretti da Lars von Trier.

Soluzione:

```
SELECT DISTINCT film.titolo
FROM film JOIN persona ON film.id_regista = persona.id_persona
WHERE persona.cognome = 'von Trier' AND
persona.nome = 'Lars';
```

- (13) Elencare i film (titoli) proiettati al cinema *Le Fontanelle*.

Soluzione:

```
SELECT DISTINCT titolo FROM film, proiezione, cinema
WHERE film.id_film = proiezione.id_film AND
proiezione.id_cinema = cinema.id_cinema AND
LOWER(cinema.nome) = 'le fontanelle';
```

- (14) Elencare i nomi ed i cognomi dei registi.

Soluzione:

```
SELECT DISTINCT nome, cognome FROM film, persona
WHERE film.id_regista = persona.id_persona;
```

- (15) Selezionare i nomi ed i cognomi degli attori.

Soluzione:

```
SELECT DISTINCT nome, cognome FROM proiezione, persona
WHERE proiezione.id_attore = persona.id_persona;
```

- (16) Selezionare i nomi ed i cognomi degli attori che sono anche registi.

Soluzione:

```
SELECT DISTINCT nome, cognome
FROM persona JOIN film ON persona.id_persona = film.id_regista
WHERE film.genere = 'drammatico'
INTERSECT
SELECT DISTINCT persona.*
FROM persona JOIN film ON persona.id_persona = film.id_regista
WHERE film.genere = 'commedia';
```

- (17) Quali attori hanno preso parte a film proiettati al cinema *Le Fontanelle* dopo l'anno 2000?

Soluzione:

```
SELECT DISTINCT persona.*
FROM cinema, proiezione, persona, partecipazione
WHERE persona.id_persona = partecipazione.id_attore AND
      proiezione.id_film = partecipazione.id_film AND
      proiezione.id_cinema = cinema.id_cinema AND
      LOWER(cinema.nome) = 'le fontanelle' AND
      proiezione.giorno > '31/12/2002';
```

- (18) Quali sono i titoli dei film interpretati da Nicole Kidman che sono stati proiettati al cinema *Le Fontanelle*?

Soluzione:

```
SELECT DISTINCT film.titolo
FROM cinema, proiezione, persona, partecipazione, film
WHERE persona.id_persona = partecipazione.id_attore AND
      film.id_film = partecipazione.id_film AND
      proiezione.id_film = film.id_film AND
      proiezione.id_cinema = cinema.id_cinema AND
      persona.nome = 'Nicole' AND
      persona.cognome = 'Kidman' AND
      LOWER(cinema.nome) = 'le fontanelle';
```

- (19) Quali sono i titoli dei film interpretati da Nicole Kidman che non sono stati proiettati al cinema *Le Fontanelle*?

Soluzione:

```
SELECT DISTINCT film.titolo
FROM persona, partecipazione, film
WHERE persona.id_persona = partecipazione.id_attore AND
      film.id_film = partecipazione.id_film AND
      persona.nome = 'Nicole' AND
      persona.cognome = 'Kidman';
EXCEPT
SELECT DISTINCT film.titolo
FROM cinema, proiezione, film
WHERE proiezione.id_film = film.id_film AND
```

```
    proiezione.id_cinema = cinema.id_cinema AND  
    LOWER(cinema.nome) = 'le fontanelle';
```

- (20) Quali sono i cognomi delle persone che contengono la lettera *s*?

Soluzione:

```
SELECT DISTINCT cognome  
FROM persona  
WHERE cognome LIKE '%s%';
```

- (21) Elencare i titoli dei film che contengono almeno tre *e*.

Soluzione:

```
SELECT titolo  
FROM film  
WHERE titolo LIKE '%e%%e%%e%';
```