

SECONDO PARZIALE - Basi di Dati e Lab

24/06/2019

Corso di Laurea in Ingegneria Informatica

Prof. Sonia Bergamaschi

Esercizio 1 (punti 8)

È necessario totalizzare almeno 4 punti in questo esercizio affinché il resto della prova venga valutato

- Operatori di JOIN dell'algebra relazionale.
- Istruzione SQL GROUP BY.

Esercizio 2 (punti 25)

Sia dato il seguente schema relazionale relativo alla gestione dell'organizzazione di concerti.

Artista (nome_artista, tipo, anni_attività)

Sede (id_sede, città, num_posti)

Concerto (id_concerto, nome_artista, id_sede, data, prezzo_biglietto)

AK: nome_artista, id_sede, data

FK: nome_artista **REFERENCES** Artista

FK: id_sede **REFERENCES** Sede

Persona (CF, nome, cognome, nazionalità)

Partecipa (CF, id_concerto)

FK: CF **REFERENCES** Persona

FK: id_concerto **REFERENCES** Concerto

Si richiede di scrivere in algebra relazionale e in SQL le seguenti interrogazioni:

- Selezionare i dati delle persone di nazionalità "italiana" che hanno partecipato a concerti nella città di "Bologna". **(punti 2+2)**
- Selezionare nome e cognome delle persone che hanno partecipato solo a concerti di tipo "Rock". **(punti 2+2)**
- Selezionare gli artisti con più di 10 anni di attività che hanno tenuto concerti in tutte le sedi con più di 10000 posti nell'ultimo semestre del 2018. **(punti 2+2)**

Si richiede di scrivere in SQL la seguente interrogazione (può essere risolta anche con più viste):

- Selezionare i concerti del 2018 a cui hanno partecipato almeno 500 persone di nazionalità "italiana". **(punti 3)**
- Selezionare nome e cognome della persona che ha speso più soldi (somma dei prezzi dei biglietti) per partecipare ai concerti nel 2018. **(punti 3)**
- Creare una vista che mostri per ogni artista nome dell'artista, l'incasso totale di tutti i suoi concerti, il numero di persone e il numero di nazionalità distinte dei partecipanti ai suoi concerti. Devono essere mostrati solo gli artisti con un incasso superiore a 1000000€. **(punti 3)**
- Selezionare per ogni sede ed anno il nome dell'artista che ha tenuto il concerto con il maggior numero di partecipanti. **(punti 4)**

Esercizio 2

1. Selezionare i dati delle persone di nazionalità “italiana” che hanno partecipato a concerti nella città di “Bologna”.

$$\sigma_{\text{nazionalità}='italiana'}(Persona) \bowtie Partecipa \bowtie Concerto \bowtie \sigma_{\text{città}='Bologna'}(Sede)$$

```
SELECT *
FROM persona
WHERE nazionalità = 'italiana'
AND CF IN (
    SELECT CF
    FROM partecipa p
    JOIN concerto c ON p.id_concerto = c.id_concerto
    JOIN sede s ON s.id_sede = c.id_sede
    WHERE s.città = 'Bologna'
)
```

2. Selezionare nome e cognome delle persone che hanno partecipato solo a concerti di artisti di tipo “Rock”. (punti 2+2)

$$\begin{aligned} S1 &= \pi_{CF}(Persona \bowtie Partecipa \bowtie \sigma_{\text{tipo}='Rock'}(Artista)) \\ S2 &= \pi_{CF}(Persona \bowtie Partecipa \bowtie \sigma_{\text{tipo} <> 'Rock'}(Artista)) \\ &\pi_{\text{nome,cognome}}(Persona \bowtie (S1 - S2)) \end{aligned}$$

```
SELECT DISTINCT P.nome, P.cognome
FROM Persona P
JOIN Partecipa Pa ON P.CF = Pa.CF
JOIN Artista A ON Pa.nome_artista = A.nome_artista
WHERE A.tipo = 'Rock'
AND P.CF NOT IN (
    SELECT P1.CF
    FROM Persona P1
    JOIN Partecipa Pa1 ON P1.CF = Pa1.CF
    JOIN Artista A1 ON Pa1.nome_artista = A1.nome_artista
    WHERE A1.tipo <> 'Rock'
)
```

3. Selezionare gli artisti con più di 10 anni di attività che hanno tenuto concerti in tutte le sedi con più di 10000 posti nell'ultimo semestre del 2018. (punti 2+2)

$$\begin{aligned} S1 &= \pi_{id_sede} \left(\sigma_{posti > 10000}(Sede) \right) \\ S2 &= \sigma_{anni_attivit\grave{a} > 10}(Artista) \\ S3 &= \pi_{id_sede, nome_artista} \left(\sigma_{data \geq '1-7-2018' \wedge data \leq '31-12-2018'}(Concerto) \bowtie S2 \right) \\ S3 &\div S1 \end{aligned}$$

Selezionare gli artisti con più di 10 anni di attività per cui non esiste una sede con più di 10000 posti in cui non hanno tenuto un concerto nell'ultimo semestre del 2018

```
SELECT *
FROM artista a
WHERE anni_attività > 10
AND NOT EXISTS (
    SELECT id_sede
    FROM sede s
    WHERE num_posti > 10000
    AND NOT EXISTS (
        SELECT *
        FROM concerto c
        WHERE c.nome_artista = a.nome_artista
        AND c.id_sede = s.id_sede
        AND c.data >= '1-7-2018'
        AND c.data <= '31-12-2018'
    )
)
```

4. Selezionare i concerti del 2018 a cui hanno partecipato almeno 500 persone di nazionalità "italiana". (punti 3)

```
SELECT c.*
FROM concerto c
JOIN partecipa pa ON c.id_concerto = pa.id_concerto
JOIN persona p ON p.CF = pa.CF
WHERE p.nazionalità = 'italiana'
GROUP BY c.id_concerto
HAVING COUNT(*) >= 500
```

5. Selezionare nome e cognome della persona che ha speso più soldi (somma dei prezzi dei biglietti) per partecipare ai concerti. **(punti 3)**

```
SELECT nome, cognome
FROM persona p
JOIN partecipa pa ON p.CF = pa.CF
JOIN concerto c ON c.id_concerto = pa.id_concerto
WHERE YEAR(c.data) = 2018
GROUP BY p.CF, p.nome, p.cognome
HAVING SUM(prezzo_biglietto) >= ALL(
    SELECT SUM(prezzo_biglietto)
    FROM concerto c1
    JOIN partecipa pa1 ON c1.id_concerto = pa1.id_concerto
    WHERE YEAR(c1.data) = 2018
    GROUP BY pa1.CF
)
```

6. Creare una vista che mostri per ogni artista nome dell'artista, l'incasso totale di tutti i suoi concerti, il numero di persone e il numero di nazionalità distinte dei partecipanti ai suoi concerti. Devono essere mostrati solo gli artisti con un incasso superiore a 1000000€. **(punti 3)**

```
CREATE VIEW incasso_totale_artista AS
SELECT C.nome_artista, SUM(C.prezzo_biglietto) AS incasso_totale,
COUNT(DISTINCT nazionalità) AS num_nazionalità
FROM Concerto C
JOIN Partecipa Pa ON C.id_concerto = Pa.id_concerto
JOIN Persona P ON P.CF = Pa.CF
GROUP BY C.nome_artista
HAVING SUM(C.prezzo_biglietto) > 1000000
```

7. Selezionare per ogni sede ed anno il nome dell'artista che ha tenuto il concerto con il maggior numero di partecipanti. **(punti 4)**

```
SELECT id_sede, YEAR(data) AS anno, nome_artista
FROM concerto c
JOIN partecipa pa ON pa.id_concerto = c.id_concerto
GROUP BY id_concerto, id_sede, nome_artista, YEAR(data)
HAVING COUNT(*) >= ALL(
    SELECT COUNT(*)
    FROM concerto c1
    JOIN partecipa pa1 ON c1.id_concerto = pa1.id_concerto
    WHERE c1.id_sede = c.id_sede
    AND YEAR(c1.data) = YEAR(c.data)
    GROUP BY c1.id_concerto
)
```