

Atto 1

Algebra

Esercizio 3: Algebra Relazionale & SQL (7 punti)

Si consideri la seguente base di dati con le relazioni:

- **Biblioteca** (CodiceBiblio, Citta, Inizio) che memorizza, per le varie biblioteche, la città in cui si trova, l'anno di inizio delle attività.
- **Frequentazione** (CodiceBiblio, CFPersona) che memorizza, per ogni codice di biblioteca, il codice fiscale delle persone che hanno visitato quella biblioteca almeno una volta.

A. Nel riquadro, scrivere una Query in Algebra Relazionale che restituisce le lista dei codici fiscali delle persone che hanno visitato almeno due biblioteche. (2 punti).²

Soluzione corretta:

$F1 = Frequentazione$

$F2 = Frequentazione$

$\pi_{F1.CFPersona}(F1 \bowtie_{F1.CodiceBiblio <> F2.CodiceBiblio \text{ AND } F1.CFPersona = F2.CFPersona} F2)$

Algebra aggiuntiva

Algebra in più

Esercizi Base

Esercizio 1 - Biblioteche fondate dopo il 2000:

```
 $\sigma(\text{Inizio} > 2000)(\text{Biblioteca})$ 
```

Esercizio 2 - Città con biblioteche:

```
 $\pi(\text{Citta})(\text{Biblioteca})$ 
```

Esercizio 3 - Persone che frequentano biblioteche di Roma:

```
 $\pi(\text{CFPersona})(\sigma(\text{Citta} = \text{'Roma'})(\text{Biblioteca}) \bowtie \text{Frequentazione})$ 
```

Esercizi Intermedi

Esercizio 4 - Persone che NON frequentano biblioteche:

$$\pi(\text{CFPersona})(\text{Persone}) - \pi(\text{CFPersona})(\text{Frequentazione})$$

Nota: assumo l'esistenza di una relazione Persone, altrimenti non è esprimibile

Esercizio 5 - Persone che frequentano almeno 2 biblioteche:

```
F1 ← Frequentazione
F2 ← Frequentazione

$$\pi(\text{F1.CFPersone})(\text{F1} \bowtie (\text{F1.CFPersone} = \text{F2.CFPersone} \wedge \text{F1.CodiceBiblio} \neq \text{F2.CodiceBiblio}) \text{ F2})$$

```

Esercizio 6 - Città con biblioteche pre-1990 non frequentate:

```
BiblioVecchie ←  $\sigma(\text{Inizio} < 1990)(\text{Biblioteca})$ 
BiblioFrequentate ←  $\pi(\text{CodiceBiblio})(\text{Frequentazione})$ 
BiblioNonFrequentate ←  $\pi(\text{CodiceBiblio})(\text{BiblioVecchie}) - \text{BiblioFrequentate}$ 
 $\pi(\text{Citta})(\text{BiblioVecchie} \bowtie \rho(\text{CodiceBiblio2} \leftarrow \text{CodiceBiblio})(\text{BiblioNonFrequentate}))$ 
```

SQL

B. Scrivere una query in Standard SQL per restituire le città in cui tutte le biblioteche che hanno iniziato l'attività nello stesso anno. Per esempio, una città "X" è restituita, se tutte le biblioteche in "X" hanno iniziato le loro attività nello stesso anno. (2.5 punti).

```
SELECT DISTINCT B1.Citta
FROM Biblioteca B1
JOIN Biblioteca B2 ON B1.CodiceBiblio = B2.CodiceBiblio
WHERE B1.Inizio = B2.Inizio
AND B1.CodiceBiblio = B2.CodiceBiblio;
```

C. Nel riquadro, scrivere una query in Standard SQL che restituisce le biblioteche con un numero di frequentatori più alto della media tra tutte le biblioteche (2.5 punti)

```
CREATE VIEW Conteggio_visite AS
SELECT COUNT(*) AS N_Visitatori, CodiceBiblio
```

```

FROM Frequentazione
GROUP BY CodiceBiblio;

CREATE VIEW Media_conteggio_visite AS
SELECT AVG(N_visitatori) AS Media_visite, CodiceBiblio
FROM Frequentazione F
JOIN Conteggio_visite CV ON F.CodiceBiblio = CV.CodiceBiblio
GROUP BY CodiceBiblio;

CREATE VIEW Max_visite AS
SELECT CodiceBiblio
FROM Media_conteggio_visite
WHERE Media_visite = (SELECT MAX(Media_visite) FROM Media_conteggio_visite);

```

Esercizio 4: Normalizzazione (5 punti)

Sia data la seguente relazione $R(ABCDE)$, con copertura ridotta $G=\{B \rightarrow C, B \rightarrow E, C \rightarrow B, A \rightarrow B, A \rightarrow D\}$. Risolvere i seguenti punti:

- Trovare la/e chiave/i di R , motivando la risposta.
- Effettuare una decomposizione in 3NF ed indicare le chiavi delle relazioni finali ottenute.
- Indicare se la decomposizione ottenuta al punto b è anche in BCNF rispetto all'insieme di dipendenze in G . Motivare la risposta.

(a) Chiavi

$$B^+ = \{B, C, E\}$$

$$C^+ = \{C, B, E\}$$

$$A^+ = \{A, B, C, D, E\}$$

A è l'unica chiave.

(b) 3FN

b.1. Sottoinsiemi

G è partizionato in sottoinsiemi tali che due dip. funz. $X \rightarrow A$ e $Y \rightarrow B$ sono insieme se $X^+ = Y^+$.

= Metto insieme i gruppi

- $B \rightarrow C, B \rightarrow E, C \rightarrow B$
 - $B^+ = \{B, C, E\}$
 - $C^+ = \{C, B, E\}$

- $A \rightarrow B, A \rightarrow D$
 - $A^+ = \{A, B, C, D, E\}$

b.2. Relazione per ogni sottoinsieme

- $R1 \{B \rightarrow C, B \rightarrow E, C \rightarrow B\}$
- $R2 \{A \rightarrow B, A \rightarrow D\}$

b.3 Se una è sotto l'altra, togliila

Se esistono due relazioni $S(X), T(Y)$ con $X \subseteq Y$, S viene eliminata.

NOPE.

b.4. Se esiste una chiave K per quale non esiste una relazione che contiene tutti gli attributi di K ,
viene aggiunta una relazione $T(K)$.

NOPE.

b.5. Indicare le chiavi ottenute dalla normalizzazione

- $R1 \{B \rightarrow C, B \rightarrow E, C \rightarrow B\}$
 - B e C chiavi
- $R2 \{A \rightarrow B, A \rightarrow D\}$
 - A chiave

c. Indicare se la decomposizione ottenuta al punto b è anche in BCNF rispetto all'insieme di dipendenze in G . Motivare la risposta.

BCNF --> OK

Spezza relazioni in più se le chiavi stanno in piedi da sole.

Nel nostro caso --> NULLA DA FARE.

Esempio di altra relazione NON in BCNF:

- $R1 \{B \rightarrow C, B \rightarrow E, C \rightarrow A\}$
 - B e C chiavi
- $R2 \{A \rightarrow B, A \rightarrow D\}$
 - A chiave

Ti rendi conto che puoi spezzare $R3 \{C \rightarrow A\}$ perché "sta in piedi da sola" e sarebbe ridondante rispetto a B .

Spiegazione Claudica 3FN - BCNF

BCNF vs 3NF: Spiegazione Tecnica ma Umana

Immagina le forme normali come dei "livelli di ordine" per organizzare i dati in un database. È come mettere in ordine la tua camera: ci sono diversi livelli di "sistemazione".

La Terza Forma Normale (3NF)

La 3NF dice: **"Ogni informazione deve dipendere direttamente dalla chiave principale, non da altre informazioni"**.

Esempio pratico:

```
Studente(Matricola, Nome, Città, CAP, Provincia)
```

Qui abbiamo un problema: la Provincia dipende dal CAP, non dalla Matricola dello studente. È come se nella tua agenda scrivessi "Mario Rossi abita a Milano, 20100, Lombardia" - ma la Lombardia dipende dal CAP 20100, non da Mario Rossi!

Per essere in 3NF: eliminiamo queste dipendenze "indirette". Separiamo le informazioni:

```
Studente(Matricola, Nome, CAP)
Località(CAP, Città, Provincia)
```

La Forma Normale di Boyce-Codd (BCNF)

La BCNF è **più severa**: dice che **"SOLO le chiavi possono determinare altre informazioni"**.

Esempio del problema:

```
Corso(Studente, Materia, Professore)
```

Dove:

- Ogni studente può seguire più materie
- Ogni materia può essere insegnata da più professori
- **MA:** ogni professore insegna solo UNA materia specifica

Qui il problema è sottile: il Professore determina la Materia (se so che c'è il Prof. Bianchi, so che insegna Matematica), ma Professore non è una chiave della tabella!

La 3NF direbbe: "Va bene così, perché Professore non è un attributo normale ma fa parte del contesto"

La BCNF dice: "No! Solo le chiavi possono determinare altri attributi. Il fatto che Bianchi→Matematica crea problemi"

La Differenza Pratica

3NF è più permissiva: accetta alcune situazioni "scomode" pur di mantenere tutte le dipendenze funzionali originali.

BCNF è più rigorosa: elimina TUTTI i problemi di dipendenza, anche se questo significa perdere alcune informazioni nel processo di decomposizione.

Quando Usare Cosa?

- **3NF:** Quando vuoi essere sicuro di non perdere nessuna informazione importante (preserva sempre le dipendenze)
- **BCNF:** Quando vuoi la massima "pulizia" dei dati, anche accettando di dover ricostruire alcune informazioni con join più complessi

Analogia finale: È come scegliere tra tenere la camera "abbastanza ordinata" (3NF) o "perfettamente ordinata" (BCNF). Nel secondo caso potresti dover mettere alcune cose in cassetti separati, ma l'ordine sarà impeccabile.

Quiz

Data la relazione $R(A, B, C)$ con una sola dipendenza funzionale $C \rightarrow A$. Una delle seguenti affermazioni è vera. Quale?

1. BC non è né chiave né superchiave
2. BC è chiave ma non è superchiave
3. BC non è chiave ma è superchiave
4. BC è sia chiave che superchiave

Dato il seguente schedule nel log fino ad un guasto

```
..., CHECKPOINT(T1), BEGIN_TR(T2), ..., BEGIN_TR(T3), ..., COMMIT(T1), ..., COMMIT (T3), Guasto
```

Vengono omessi update, insert and delete per leggibilità e perché non rilevanti alla domanda. Di quale/i transazione/i occorre fare il REDO?

1. T2
2. T2 e T1
3. T1 e T3
4. T3