



UNIVERSITÀ DEGLI STUDI DI SALERNO



Fondamenti di Informatica

Ripasso Argomenti Basi di Dati

Prof. Christian Esposito

Corso di Laurea in Ingegneria Meccanica e Gestionale (Classe I)

A.A. 2017/18

Esercizio 1

Si assuma il seguente schema relazionale per la gestione di una biblioteca:

LIBRI(codice_libro, autore, titolo)

UTENTI(codice_utente, nome, cognome)

PRESTITI(codice_utente, codice_libro, data_prestito)

Si esprimano in algebra relazionale le seguenti interrogazioni:

- Titoli dei libri presi in prestito il giorno 13/7/2011;
- Autori dei libri presi in prestito da Paolo Bianchi;
- Codici degli utenti che hanno preso in prestito libri scritti da Camilleri oppure da De Luca;
- Titoli dei libri presi in prestito il giorno 13/7/2011 da Paolo Bianchi;
- Trovare gli utenti che hanno preso in prestito solo libri di Camilleri.

Soluzione

- Autore e titolo dei cd noleggiati da Paolo Bianchi in data 20/1/2012

$S1 = \sigma_{data_prestito=13/7/2011}(Prestiti)$
 $\pi_{titolo}(S1 \bowtie Libri)$

- Autori dei libri presi in prestito da Paolo Bianchi

$S1 = \sigma_{nome='Paolo' \text{ AND } cognome='Bianchi'}(Utenti)$
 $S2 = S1 \bowtie Prestiti$
 $S3 = S2 \bowtie Libri$
 $\pi_{autore}(S3)$

- Codici degli utenti che hanno preso in prestito libri scritti da Camilleri oppure da De Luca

$S1 = \sigma_{autore='Camilleri' \text{ OR } autore='De Luca'}(Libri)$
 $S2 = S1 \bowtie Prestiti$
 $\pi_{codice_utente}(S2)$

Esercizio 1

Si assuma il seguente schema relazionale per la gestione di una biblioteca:

LIBRI(codice_libro, autore, titolo)

UTENTI(codice_utente, nome, cognome)

PRESTITI(codice_utente, codice_libro, data_prestito)

Si esprimano in algebra relazionale le seguenti interrogazioni:

- Titoli dei libri presi in prestito il giorno 13/7/2011;
- Autori dei libri presi in prestito da Paolo Bianchi;
- Codici degli utenti che hanno preso in prestito libri scritti da Camilleri oppure da De Luca;
- Titoli dei libri presi in prestito il giorno 13/7/2011 da Paolo Bianchi;
- Trovare gli utenti che hanno preso in prestito solo libri di Camilleri.

Soluzione

- d) Titoli dei libri presi in prestito il giorno 13/7/2011 da Paolo Bianchi

$S1 = \sigma_{\text{nome} = \text{'Paolo'} \text{ AND } \text{cognome} = \text{'Bianchi'}}(\text{Utenti})$

$S2 = \sigma_{\text{data_prestito} = \text{13/7/2011}}(\text{Prestiti})$

$S3 = \pi_{\text{codice_libro}}(S1 \bowtie S2)$

$S4 = S3 \bowtie \text{Libri}$

$\pi_{\text{titolo}}(S4)$

- e) Trovare gli utenti che hanno preso in prestito solo libri di Camilleri

$S1 = \sigma_{\text{autore} = \text{'Camilleri'}}(\text{Libri})$

$S2 = \pi_{\text{codice_libro}}(S1)$ $S3 = S2 \bowtie \text{Prestiti}$

$S4 = \pi_{\text{codice_utente}}(S3)$

$T1 = \sigma_{\text{autore} \neq \text{'Camilleri'}}(\text{Libri})$

$T2 = \pi_{\text{codice_libro}}(T1)$

$T3 = T2 \bowtie \text{Prestiti}$ $T4 = \pi_{\text{codice_utente}}(T3)$

$S4 - T4$

Esercizio 2

Si assuma il seguente schema relazionale per la gestione del noleggio di cd:

CD(codice_cd, autore, titolo)

CLIENTI(codice_cliente, codice_fiscale, nome, cognome)

NOLEGGIO(codice_cliente, codice_cd, data_noleggio)

Si esprimano in algebra relazionale le seguenti interrogazioni:

- Autore e titolo dei cd noleggiati da Paolo Bianchi in data 20/1/2012;
- Nome e cognome dei clienti che hanno noleggiato cd dei Radiohead in data 12/10/2001;
- Titolo dei cd che sono stati noleggiati dal cliente avente codice C123 oppure dal cliente avente codice G624;
- Trovare i clienti che hanno stesso nome e cognome, ma diverso Codice Fiscale.

Soluzione

- a) Autore e titolo dei cd noleggiati da Paolo Bianchi in data 20/1/2012

$S1 = \text{Noleggio} \bowtie \text{Clienti}$

$S2 = \sigma_{\text{data_noleggio} = 20/1/2012 \text{ AND nome} = \text{'Paolo'} \text{ AND cognome} = \text{'Bianchi'}}(S1)$

$S3 = \pi_{\text{codice_cd}}(S2)$

$\pi_{\text{autore, titolo}}(S3 \bowtie \text{CD})$

- b) Nome e cognome dei clienti che hanno noleggiato cd dei Radiohead in data 12/10/2001

$S1 = \sigma_{\text{data_noleggio} = 12/10/2001}(\text{Noleggio})$

$S2 = \sigma_{\text{autore} = \text{'Radiohead'}}(\text{CD})$

$S3 = \pi_{\text{codice_cliente}}(S1 \bowtie S2)$

$\pi_{\text{nome, cognome}}(S3 \bowtie \text{Clienti})$

Esercizio 2

Si assuma il seguente schema relazionale per la gestione del noleggio di cd:

CD(codice_cd, autore, titolo)

CLIENTI(codice_cliente, codice_fiscale, nome, cognome)

NOLEGGIO(codice_cliente, codice_cd, data_noleggio)

Si esprimano in algebra relazionale le seguenti interrogazioni:

- Autore e titolo dei cd noleggiati da Paolo Bianchi in data 20/1/2012;
- Nome e cognome dei clienti che hanno noleggiato cd dei Radiohead in data 12/10/2001;
- Titolo dei cd che sono stati noleggiati dal cliente avente codice C123 oppure dal cliente avente codice G624;
- Trovare i clienti che hanno stesso nome e cognome, ma diverso Codice Fiscale.

Soluzione

- Titolo dei cd che sono stati noleggiati dal cliente avente codice C123 oppure dal cliente avente codice G624

$$\pi_{\text{titolo}} (\sigma_{\text{codice_cliente} = \text{'C123'} \text{ OR } \text{codice_cliente} = \text{'G624'}} (\text{Noleggio}) \bowtie \text{CD})$$

- Trovare i clienti che hanno stesso nome e cognome, ma diverso Codice Fiscale

$$\text{Clienti1} = \rho_{\text{nome1} \leftarrow \text{nome}, \text{cognome1} \leftarrow \text{cognome}, \text{codice_fiscale1} \leftarrow \text{codice_fiscale}} (\text{Clienti})$$

$$\pi_{\text{nome}, \text{cognome}, \text{codice_fiscale}} (\text{Clienti} \bowtie_{\text{Clienti.nome} = \text{Clienti1.nome1 AND Clienti.cognome} = \text{Clienti1.cognome1 AND Clienti.codice_fiscale} \neq \text{Clienti1.codice_fiscale1}} \text{Clienti1})$$

Esercizio 5 Si assuma il seguente schema relazionale per la gestione degli affitti:
APPARTAMENTO(CodA, Superficie, Indirizzo, Citta)
CONTRATTO-AFFITTO(CodA, DataInizio, DataFine, NomePersona, RettaMensile)

Si esprimano come query SQL le seguenti interrogazioni:

- Trovare il nome delle persone che hanno stipulato più di due contratti di affitto per lo stesso appartamento (in tempi diversi);
- Trovare il codice e l'indirizzo degli appartamenti di Torino in cui la retta mensile è sempre stata superiore a 500 euro e per cui sono stati stipulati al più 5 contratti;
- Trovare il codice e l'indirizzo degli appartamenti di Torino per cui sono stati stipulati al più 5 contratti (un numero variabile tra 0 e 5).

Soluzione

- Trovare il nome delle persone che hanno stipulato più di due contratti di affitto per lo stesso appartamento (in tempi diversi)

```
SELECT NomePersona FROM CONTRATTO-AFFITTO  
GROUP BY CodA, NomePersona  
HAVING COUNT(*) > 2;
```
- Trovare il codice e l'indirizzo degli appartamenti di Torino in cui la retta mensile è sempre stata superiore a 500 euro e per cui sono stati stipulati al più 5 contratti

```
SELECT A.CodA, Indirizzo FROM APPARTAMENTO A, CONTRATTO-AFFITTO CA  
WHERE A.CodA=CA.CodA AND Citta='Torino' AND A.CodA NOT IN  
(SELECT CodA FROM CONTRATTO-AFFITTO WHERE RettaMensile <= 500)  
GROUP BY A.CodA, Indirizzo  
HAVING COUNT(*) <= 5;
```

Esercizio 5 Si assuma il seguente schema relazionale per la gestione degli affitti:
APPARTAMENTO(CodA, Superficie, Indirizzo, Citta)
CONTRATTO-AFFITTO(CodA, DataInizio, DataFine, NomePersona, RettaMensile)

Si esprimano come query SQL le seguenti interrogazioni:

- Trovare il nome delle persone che hanno stipulato più di due contratti di affitto per lo stesso appartamento (in tempi diversi);
- Trovare il codice e l'indirizzo degli appartamenti di Torino in cui la retta mensile è sempre stata superiore a 500 euro e per cui sono stati stipulati al più 5 contratti;
- Trovare il codice e l'indirizzo degli appartamenti di Torino per cui sono stati stipulati al più 5 contratti (un numero variabile tra 0 e 5).

Soluzione

- Trovare il codice e l'indirizzo degli appartamenti di Torino in cui la retta mensile è sempre stata superiore a 500 euro e per cui sono stati stipulati al più 5 contratti

```
SELECT A.CodA,Indirizzo FROM APPARTAMENTO A, CONTRATTO-AFFITTO CA
WHERE A.CodA=CA.CodA AND Citta='Torino'
GROUP BY A.CodA, Indirizzo
HAVING COUNT(*)<=5
AND MIN(RettaMensile)>500;
```
- Trovare il codice e l'indirizzo degli appartamenti di Torino per cui sono stati stipulati al più 5 contratti (un numero variabile tra 0 e 5).

```
SELECT CodA,Indirizzo FROM APPARTAMENTO
WHERE Citta='Torino' AND CodA NOT IN
(SELECT CodA FROM CONTRATTO-AFFITTO GROUP BY CodA HAVING COUNT(*)>5);
```

Esercizio 6

Si assuma il seguente schema relazionale per la gestione di gare sportive:

GARA(CodG, Luogo, Data, Disciplina)

ATLETA(CodA, Nome, Nazione, DataNascita)

PARTECIPAZIONE(CodG, CodA, PosizioneArrivo, Tempo)

Si esprimano come query SQL le seguenti interrogazioni:

- a) Trovare il nome e la data di nascita degli atleti italiani che non hanno partecipato a nessuna gara di discesa libera;
- b) Trovare le nazioni per cui concorrono almeno 5 atleti nati prima del 1980, ciascuno dei quali abbia partecipato ad almeno 10 gare di sci di fondo.

Soluzione

- a) Trovare il nome e la data di nascita degli atleti italiani che non hanno partecipato a nessuna gara di discesa libera

```
SELECT Nome, DataNascita FROM ATLETA
WHERE Nazione='Italia' AND CodA NOT IN
(SELECT CodA FROM PARTECIPAZIONE P, GARA G
WHERE P.CodG=G.CodG AND G.Disciplina='discesa libera');
```

Esercizio 6

Si assuma il seguente schema relazionale per la gestione di gare sportive:

GARA(CodG, Luogo, Data, Disciplina)

ATLETA(CodA, Nome, Nazione, DataNascita)

PARTECIPAZIONE(CodG, CodA, PosizioneArrivo, Tempo)

Si esprimano come query SQL le seguenti interrogazioni:

- a) Trovare il nome e la data di nascita degli atleti italiani che non hanno partecipato a nessuna gara di discesa libera;
- b) Trovare le nazioni per cui concorrono almeno 5 atleti nati prima del 1980, ciascuno dei quali abbia partecipato ad almeno 10 gare di sci di fondo.

Soluzione

- b) Trovare le nazioni per cui concorrono almeno 5 atleti nati prima del 1980, ciascuno dei quali abbia partecipato ad almeno 10 gare di sci di fondo

```
SELECT Nazione FROM ATLETA
WHERE DataNascita < '1/1/1980' AND CodA IN
(SELECT CodA FROM PARTECIPAZIONE P, GARA G
WHERE P.CodG = G.CodG AND Disciplina = 'fondo'
GROUP BY CodA HAVING COUNT(*) >= 10)
GROUP BY Nazione
HAVING COUNT(*) >= 5;
```

Esercizio 9

Si consideri la seguente relazione per libri pubblicati:

LIBRO (Titolo_Libro, Nome_Autore, Tipo_Libro, Prezzo_di_Listino, Affiliazione_Autore, Editore)

Si supponga che esistano le seguenti dipendenze:

Titolo_Libro \rightarrow Editore, Tipo_Libro

Tipo_Libro \rightarrow Prezzo_di_Listino

Nome_Autore \rightarrow Affiliazione_Autore

- a) In che forma normale è la relazione? Si motivi la risposta.
- b) Si esegua la normalizzazione finché non sia più possibile decomporre le relazioni. Si enuncino le motivazioni che giustificano ciascuna decomposizione.

Soluzione

- a) In che forma normale è la relazione? Si motivi la risposta.

La chiave della relazione è Titolo_Libro, Nome_Autore.

La relazione è in 1NF ma non in 2NF poiché non ci sono attributi che hanno una dipendenza funzionale completa rispetto alla chiave. Inoltre, la relazione non è neppure in 3NF.

- b) Si esegua la normalizzazione finché non sia più possibile decomporre le relazioni. Si enuncino le motivazioni che giustificano ciascuna decomposizione.

Decomposizione in 2NF:

- Libro0(Titolo_Libro, Nome_Autore)
- Libro1(Titolo_Libro, Editore, Tipo_Libro, Prezzo_di_Listino)
- Libro2(Nome_Autore, Affiliazione_Autore)

Questa decomposizione elimina le dipendenze parziali, ma sussiste la dipendenza transitiva di Prezzo_di_Listino.

Esercizio 9

Si consideri la seguente relazione per libri pubblicati:

LIBRO (Titolo_Libro, Nome_Autore, Tipo_Libro, Prezzo_di_Listino, Affiliazione_Autore, Editore)

Si supponga che esistano le seguenti dipendenze:

Titolo_Libro \rightarrow Editore, Tipo_Libro

Tipo_Libro \rightarrow Prezzo_di_Listino

Nome_Autore \rightarrow Affiliazione_Autore

- a) In che forma normale è la relazione? Si motivi la risposta.
- b) Si esegua la normalizzazione finché non sia più possibile decomporre le relazioni. Si enuncino le motivazioni che giustificano ciascuna decomposizione.

Soluzione

Decomposizione in 3NF:

- Libro0 (Titolo_Libro, Nome_Autore)
- Libro1-1 (Titolo_Libro, Editore, Tipo_Libro)
- Libro1-2 (Tipo_Libro, Prezzo_di_Listino)
- Libro2 (Nome_Autore, Affiliazione_Autore)

Questa decomposizione elimina la dipendenza transitiva di Prezzo_di_Listino.