

1. Esonero SQL

Le seguenti relazioni definiscono una base di dati “Twilight” per gestire un laboratorio di analisi del sangue. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

UTENTE(Nome, Cognome, Eta, Sesso)

REFERTO(Nome, Cognome, Data, Analisi, Esito)

ANALISI(Nome, Tipo, Digiuno, Prezzo, NormaMin, NormaMax)

Vincoli di integrità referenziale:

REFERTO(Nome, Cognome) referencia UTENTE(Nome, Cognome),

REFERTO(Analisi) referencia ANALISI(Nome).

Significato delle relazioni e degli attributi. *REFERTO* contiene i singoli esiti di un referto per le varie analisi effettuate in una certa data, *Data* è la data di effettuazione dell’analisi ed è rappresentata come un testo nel formato ‘YYYY/MM/DD’, *Esito* è un valore numerico che indica il risultato dell’analisi. *ANALISI* contiene le possibili analisi effettuate nel laboratorio: *Tipo* può assumere i valori *ematologia*, *endocrinologia*, *biochimica*, *coagulazione*; *Digiuno* indica è un booleano che indica se è richiesto il digiuno; *NormaMin* e *NormaMax* indicano l’intervallo dei valori nella norma per un esito. *Eta* può essere NULL quando non si conosce l’età dell’utente. Le rimanenti relazioni e attributi sono autoesplicativi.

Esprimere in SQL tutte e tre le seguenti interrogazioni indicando a quale query si sta rispondendo (A, B, C). Commentare inoltre la soluzione proposta.

(A - Bassa complessità) Elencare, senza duplicati e ordinati per cognome e nome, gli utenti maschi tra 20 e 40 anni e femmine tra 50 e 70 anni che (in entrambi i casi) hanno almeno un’analisi fuori dalla norma.

(B - Media complessità) Trovare, tra le analisi che costano almeno 100 euro, quelle che nei mesi di gennaio del decennio 2010-2019 sono state eseguite complessivamente almeno 200 volte. Mostrare il nome dell’analisi, il numero di volte in cui è stata effettuata, il prezzo medio e il prezzo totale. Non usare sottoquery.

(C - Alta complessità) Trovare il più giovane utente maschio che ha avuto esclusivamente analisi nella norma. Nel risultato non devono comparire eventuali utenti che non hanno effettuato nessuna analisi. Non usare costrutti come ROWNUM, TOP, LIMIT.

Soluzioni.

A.

```
select distinct u.cognome, u.nome
from utente u join referto r on u.nome=r.nome and u.cognome=r.cognome
join analisi a on r.analisi=a.nome
where ((sesso='M' and eta>=20 and eta<=40) or (sesso='F' and eta>=50 and eta<=70)) and
(esito < normamin or esito > normamax)
order by u.cognome, u.nome;
```

Commento:

Da analisi si ricavano i valori della norma, da referto i valori effettivi e da utente i dati anagrafici dell’utente. Si selezionano gli utenti con valore al di sotto del valore minimo o al di sopra del valore massimo e che siano o maschi delle età indicata o femmine delle età indicate. Tramite order by si dà ordinamento per cognome e nome e tramite distinct si rimuovono eventuali duplicati.

B.

```
select distinct a.nome, count(*), prezzo, sum(prezzo)
from utente u join referto r on u.nome=r.nome and u.cognome=r.cognome
join analisi a on r.analisi=a.nome
where data like '201_/01/_' and prezzo >= 100
```

```
group by a.nome  
having count(*) >= 200;
```

Commento:

Si ricavano le analisi del periodo indicato tramite l'operatore like con caratteri jolly sulla stringa che riporta la data. Si raggruppano le analisi per nome per contare quante ne sono state effettuate e tramite having tenere solo quelle effettuate più di 200 volte. Si calcolano con gli operatori aggregati il numero e il ricavo totale. Per il prezzo non occorre operatore aggregato per la dipendenza funzionale tra la chiave primaria di analisi (secondo cui raggruppiamo) e l'attributo prezzo.

C.

```
with utentifuorinorma as (  
select distinct nome,cognome,eta  
from referto r join analisi a on r.analisi=a.nome  
where r.esito < normamin or r.esito > normamax),  
utentimaschisolorma as (  
select nome,cognome,eta  
from referto r join utente u on r.nome=u.nome and r.cognome=u.cognome  
where sesso='M' and (nome,cognome) not in (  
select nome,cognome  
from utentifuorinorma)  
select nome,cognome from utentimaschisolorma where eta = (  
select min(eta) from utentimaschisolorma);
```

Commento:

Con la clausola with definiamo due sottoquery: utentifuorinorma ricava gli utenti (compresa l'età) degli utenti con almeno un'analisi fuori norma, utentimaschisolorma esclude gli utenti in utentifuorinorma ricavando gli utenti maschi con solo analisi nella norma. Infine ricaviamo, tra utentimaschisolorma, l'età minima e l'utente (o eventualmente gli utenti) che hanno tale età minima.

2. Esonero Algebra

Le seguenti relazioni definiscono una base di dati "Twilight" per gestire un laboratorio di analisi del sangue. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

UTENTE(Nome, Cognome, Eta, Sesso)

REFERTO(Nome, Cognome, Data, Analisi, Esito)

ANALISI(Nome, Tipo, Diggiuno, Prezzo, NormaMin, NormaMax)

Vincoli di integrità referenziale:

REFERTO(Nome, Cognome) referencia UTENTE(Nome, Cognome),

REFERTO(Analisi) referencia ANALISI(Nome).

Significato delle relazioni e degli attributi. *REFERTO* contiene i singoli esiti di un referto per le varie analisi effettuate in una certa data, *Data* è la data di effettuazione dell'analisi ed è rappresentata come un testo nel formato 'YYYY/MM/DD', *Esito* è un valore numerico che indica il risultato dell'analisi. *ANALISI* contiene le possibili analisi effettuate nel laboratorio: *Tipo* può assumere i valori *ematologia*, *endocrinologia*, *biochimica*, *coagulazione*; *Digiuno* indica è un booleano che indica se è richiesto il digiuno; *NormaMin* e *NormaMax* indicano l'intervallo dei valori nella norma per un esito. *Eta* può essere NULL quando non si conosce l'età dell'utente. Le rimanenti relazioni e attributi sono autoesplicativi.

Esprimere tutte e due le seguenti interrogazioni indicando a quale query si sta rispondendo (per facilità di scrittura delle formule, si possono eseguire gli esercizi con carta e penna e fare l'upload delle foto prestando attenzione che lo svolgimento sia chiaramente leggibile):

(A - Algebra Relazionale) Elencare gli utenti che hanno effettuato **tutte** le analisi per cui è richiesto il digiuno. Scrivere la query sotto forma di albero sintattico.

(B - Calcolo Relazionale su tuple con dichiarazione di range) Elencare gli utenti che hanno effettuato **solo** analisi per cui è richiesto il digiuno. Nel risultato non devono comparire eventuali utenti che non hanno effettuato nessuna analisi.

Soluzioni.

A.

$$\pi_{nome,cognome,analisi}(REFERTO) \div \rho_{analisi \leftarrow nome}(\pi_{nome}(\sigma_{Digiuno=TRUE}(ANALISI)))$$

B. $\{r.nome, r.cognome | r(referto)\}$

$$\forall r'(referto)((r'.nome = r.nome \wedge r'.cognome = r.cognome) \Rightarrow (\exists a(analisi)(a.nome = r'.analisi \wedge a.digiuno = TRUE)))\}$$

3. Esonero Ottimizzazione

Le seguenti relazioni definiscono una base di dati “Twilight” per gestire un laboratorio di analisi del sangue. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

UTENTE(Nome, Cognome, Eta, Sesso)

REFERTO(Nome, Cognome, Data, Analisi, Esito)

ANALISI(Nome, Tipo, Digiuno, Prezzo, NormaMin, NormaMax)

Vincoli di integrità referenziale:

REFERTO(Nome, Cognome) referencia UTENTE(Nome, Cognome),

REFERTO(Analisi) referencia ANALISI(Nome).

Significato delle relazioni e degli attributi. *REFERTO* contiene i singoli esiti di un referto per le varie analisi effettuate in una certa data, *Data* è la data di effettuazione dell’analisi ed è rappresentata come un testo nel formato ‘YYYY/MM/DD’, *Esito* è un valore numerico che indica il risultato dell’analisi. *ANALISI* contiene le possibili analisi effettuate nel laboratorio: *Tipo* può assumere i valori *ematologia*, *endocrinologia*, *biochimica*, *coagulazione*; *Digiuno* indica è un booleano che indica se è richiesto il digiuno; *NormaMin* e *NormaMax* indicano l’intervallo dei valori nella norma per un esito. *Eta* può essere NULL quando non si conosce l’età dell’utente. Le rimanenti relazioni e attributi sono autoesplicativi.

Data la seguente query:

$$\sigma_{digiuno=False \wedge referto.nome='Edward' \wedge cognome='Cullen' \wedge (tipo='biochimica' \vee tipo='coagulazione')} (referto \bowtie_{analisi=analisi.nome} analisi)$$

disegnare gli alberi sintattici prima e dopo l’ottimizzazione logica e calcolare il numero di tuple “mosse” prima e dopo l’ottimizzazione logica.

Si svolgano i calcoli sapendo che:

$$CARD(utente) = VAL((nome,cognome),referto) = 1\ 000$$

$$CARD(referto) = 40\ 000$$

$$CARD(analisi) = VAL(analisi,referto) = 320$$

$$MIN(normamin,analisi) = 0$$

$$MAX(normamin,analisi) = 500$$

Soluzioni.

La query ottimizzata dividendo la selezione e portandola verso le foglie è

$$\sigma_{referto.nome='Edward' \wedge cognome='Cullen'}(referto) \bowtie_{analisi=analisi.nome} \sigma_{digiuno=False \wedge (tipo='biochimica' \vee tipo='coagulazione')}(analisi)$$

Prima dell’ottimizzazione:

- Costo $r_1 = referto \bowtie_{analisi=analisi.nome} analisi : 40\ 000 \cdot 320 = 12\ 800\ 000$.
- Cardinalità di $|r_1| = CARD(referto) = 40\ 000$ (equijoin attraverso la chiave esterna)
- Costo della selezione: $|r_1|$

- Costo totale = $12\,800\,000 + 40\,000 \approx 12\,800\,000$.

Dopo l'ottimizzazione:

- Costo $\sigma_1 = \sigma_{referto.nome='Edward' \wedge cognome='Cullen'}(referto) = 40\,000$
- Tuple prodotte dalla selezione $|\sigma_1| = \frac{1}{VAL((nome,cognome),referto)} \cdot CARD(referto) = \frac{1}{1000} \cdot 40\,000 = 40$
- Costo $\sigma_2 = \sigma_{digiuno=False \wedge (tipo='biochimica' \vee tipo='coagulazione')}(analisi) : CARD(analisi) = 320$
- Tuple prodotte dalla selezione $|\sigma_2| = \frac{1}{VAL(digiuno,analisi)} \cdot (1 - ((1 - \frac{1}{VAL(tipo,analisi)}) \cdot (1 - \frac{1}{VAL(tipo,analisi)}))) \cdot CARD(analisi) = \frac{1}{2} \cdot (1 - ((1 - \frac{1}{4}) \cdot (1 - \frac{1}{4}))) \cdot 320 = \frac{1}{2} \cdot (1 - (\frac{3}{4} \cdot \frac{3}{4})) \cdot 320 = \frac{1}{2} \cdot \frac{7}{16} \cdot 320 = 70$
- Costo join $r = \sigma_1 \bowtie_{analisi=analisi.nome} \sigma_2 : |\sigma_1| \cdot |\sigma_2| = 40 \cdot 70 = 2\,800$.
- Costo totale = $40\,000 + 320 + 2\,800 \approx 43\,000$

4. Esonero Teoria 3NF

Dati:

$R(A, B, C, D, E, F, G)$ e

$F = \{A E F \rightarrow G, G \rightarrow E F, F \rightarrow C D, C F \rightarrow B\}$

dire, motivando la risposta, se R è in 3FN e se non lo è decomporla in relazioni in 3FN esplicitando tutti i passaggi. Il risultato è BCNF? Perché?

Soluzione.

Per prima cosa, occorre individuare la o le chiavi della relazione R.

Ogni chiave deve contenere A, perché non compare a destra in nessuna dipendenza funzionale. La chiave non può contenere D, C e B, perché compaiono sempre e solo a destra. Una possibile superchiave è quindi: $\{A E F G\}^+ = \{A E F G | C D | B\}$, ma non è chiave, infatti i suoi sottoinsiemi $\{A G\}^+ = \{A G | E F | C D | B\}$ e $\{A E F\}^+ = \{A E F | G | C D | B\}$ sono superchiavi minimali e quindi chiavi candidate.

La relazione non è in 3NF, infatti per esempio $C F \rightarrow B$ non è riflessiva, non ha a sinistra una superchiave né a destra attributi primi.

Troviamo quindi la copertura minimale di F, partendo dalla sua forma canonica:

$F' = \{A E F \rightarrow G, G \rightarrow E, G \rightarrow F, F \rightarrow C, F \rightarrow D, C F \rightarrow B\}$

C è estraneo in $C F \rightarrow B$, quindi:

$F' = \{A E F \rightarrow G, G \rightarrow E, G \rightarrow F, F \rightarrow C, F \rightarrow D, F \rightarrow B\}$.

Non ci sono dipendenze funzionali ridondanti.

Possiamo ora scomporre in 3NF ottenendo le seguenti relazioni dopo aver accorpato le d.f. con lo stesso antecedente:

$R1(\underline{A}, E, F, G)$

$R2(\underline{G}, E, F)$

$R3(\underline{F}, B, C, D)$

R1 contiene una chiave. È ancora possibile includere R2 in R1, quindi:

$R1(\underline{A}, E, F, G)$ con $F_1 = \{G \rightarrow E F\}$

$R3(\underline{F}, B, C, D)$

Il risultato non è BCNF perché R1 contiene una dipendenza funzionale che non è né superchiave né banale.

5. Esonero Teoria Armstrong

Usando le regole di Armstrong derivare F' da F , dove $F = \{AB \rightarrow C, CD \rightarrow EF\}$ e $F' = \{ABD \rightarrow E\}$.

Soluzione.

Partendo da $F = \{AB \rightarrow C, CD \rightarrow EF\}$ applicando la regola di pseudotransitività otteniamo $ABD \rightarrow EF$

Per decomposizione otteniamo

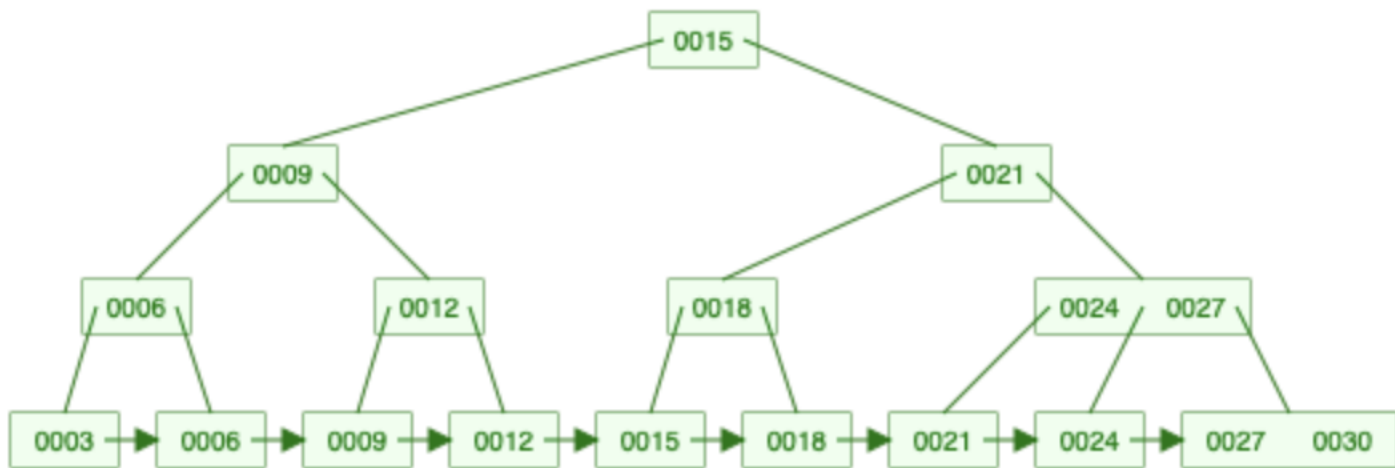
$ABD \rightarrow E$.

6. Esonero Teoria B+-Tree

Rappresentare un possibile B+-tree contenente le chiavi (3, 6, 9, 12, 15, 18, 21, 24, 27, 30) con $m=3$. Non si richiede di simulare le singole operazioni di inserimento, ma di mostrare un possibile B+-tree con le caratteristiche indicate.

Soluzione. Ogni nodo ha al massimo $m - 1 = 2$ chiavi, la radice ha almeno 1 chiave, gli altri nodi hanno almeno $\lceil \frac{m}{2} - 1 \rceil = 1$ chiave. Le foglie devono contenere tutte le chiavi e essere linkate.

Un possibile B+-tree è in figura.



7. Esonero Teoria 2PL

Considerare la seguente storia interfogliata

$S = r1(x), r1(y), r2(x), r3(y), r2(y), w2(x), w3(y)$ La storia S è compatibile con il protocollo 2PL? Giustificare la risposta.

Soluzione.

La storia S è compatibile con 2PL. Un esempio di aggiunta dei lock 2PL è il seguente:

$S = LS1(x), r1(x), LS1(y), r1(y), LS2(x), r2(x), LS3(y), r3(y), LS2(y), r2(y), UL1(x), LX2(x), w2(x), UL1(y), UL2(y), LX3(y), w3(y)$.