

1
SETTEMBRE 2016

ACID



NOME
COGNOME
MATRICOLA

ESERCIZIO 1 (12 punti)

Dato il seguente schema relazionale, che modella i dati di mailing list per studenti, associate a corsi offerti dal dipartimento di Informatica del Management.

STUDENTE(Email, Nome, Cognome, AnnoImmatricolazione)

MAILING_LIST(EmailStudente, CodCorso)

CORSO(Codice, Nome, Anno)

Con vincoli di integrità referenziale:

MAILING_LIST.EmailStudente → STUDENTE.Email

MAILING_LIST.CodCorso → CORSO.Codice

a) (2 pt) Scrivere in SQL la query che determina, per ogni studente immatricolato dopo il 1999, il numero di corsi cui è iscritto (per iscrizione si intende la partecipazione alla mailing list del corso).

b) (3 pt) Scrivere in SQL la query che determina, tra tutti i corsi del terzo anno, quello (o quelli, in caso di parità) avente/i il maggior numero di iscritti. Restituire codice e nome del corso/i in questione. [VINCOLO: Non è possibile usare l'operatore di HAVING].

c) (3 pt) Scrivere in SQL la query che determina email, nome e cognome degli studenti iscritti ad ALMENO due corsi del primo anno. [VINCOLO: Non è possibile usare l'operatore di COUNT].

d) (2 pt) Scrivere in SQL il codice della tabella STUDENTE, imponendo i seguenti vincoli: (i) Cognome deve essere sempre definito; (ii) AnnoImmatricolazione deve essere maggiore di 1980; (iii) non possono esistere più di 3 studenti con lo stesso nome e cognome.

d) (2 pt) Assumendo che STUDENTE sia una Collezione in MONGO-DB, scrivere la query MONGO-DB che restituisce email e cognome (*) degli utenti che si chiamano "Mario" OPPURE che si sono immatricolati dopo il 2000.

(*) restituire solo i campi indicati

ESERCIZIO 2 (12 punti)

Si vuole progettare una base di dati per la gestione dei circoli di scacchi della regione Emilia-Romagna. Ogni circolo dispone di un nome (univoco), un indirizzo, un recapito telefonico, un'email. Si vogliono gestire le informazioni dei soci di ciascun circolo. Ogni socio ha un numero di tessera (univoco all'interno del circolo d'appartenenza), nome, cognome, data di nascita, un indirizzo di domicilio, ed uno o più recapiti telefonici. Si vogliono memorizzare le statistiche associate a ciascun socio, quali: numero di partite giocate, vinte, perse, pareggiate, e data di iscrizione al circolo. I soci possono appartenere a tre categorie, in base al loro livello di gioco: Maestri, Matricole o Professionisti. I soci Maestri possono offrire corsi didattici per l'acquisizione di abilità nel gioco degli scacchi. Ogni corso ha una durata di inizio e fine, un numero di ore, un numero di lezioni, un programma, ed è tenuto da uno o più Maestri del circolo. Ai corsi possono iscriversi solo soci Matricole; si vuole tenere traccia delle iscrizioni di ogni corso. Infine, si vogliono gestire le informazioni relative ai tornei di scacchi organizzati da un circolo. Ogni torneo dispone di un nome, una data di inizio, una data di fine, un premio finale (in euro), ed è composto da un certo numero di partite. Ogni partita è svolta da due soci Professionisti, e dispone di data, durata, punteggio del giocatore 1, punteggio del giocatore 2. Per ogni partita, si vuole inoltre memorizzare l'elenco delle mosse svolte. Ogni mossa dispone di un numero progressivo (per quella partita) e di un codice esplicativo, ed è associata al giocatore che l'ha effettuata.

a) (6 pt) Costruire il modello Entita-Relazione (E-R) della base di dati.

b) (4 pt) Tradurre il modello E-R nel modello logico relazionale, scegliendo la soluzione che minimizza il numero di valori NULL nelle tabelle. Indicare i vincoli di integrità referenziale tra gli attributi dello schema.

c) (2pt) Indicare quale operazione ha il costo più alto, tra quelle indicate:

- Creare un nuovo circolo (interattiva, $w_1=1$, frequenza: 2 volte/mese) %
- Aggiungere una partita ad un torneo esistente (interattiva, $w_1=1$, frequenza: 5 volte/mese) %
- Visualizzare tutte le mosse svolte in una certa partita (interattiva, $w_1=1$, frequenza: 1 volta/mese) %

%

Tabella media dei volumi: 3 circoli, 10 partite per torneo, 25 mosse per partita. α (peso operazioni scrittura) = 2, β (peso operazioni lettura) = 1

ESERCIZIO 3 (4 punti)

Dato il seguente schema: $R(ABCDEF)$, con le seguenti dipendenze funzionali:

$$AB \rightarrow DE, A \rightarrow C, A \rightarrow F$$

a) Indicare se ABF sia una chiave o meno della relazione R. Giustificare le risposte.

b) La relazione e' in forma normale di Boyce e Codd (FNBC)? La relazione e' in terza forma normale (3FN)? Giustificare le risposte.

$$A_F^+ = \{A, C, F\}$$

$$AB_F^+ = \{A, B, D, E, C, F\} \rightarrow \text{CHIAVE quindi ABF non \u00e8}$$

CHIAVE, al massimo sara' SUPERCHIAVE!

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

$$AF_F^+ = \{A, F, C\}$$

$$AB_F^+ = \{A, B, D, E, C, F\} \rightarrow \text{CHIAVE}$$

$$BF_F^+ = \{B, F, \}$$

FNBC \rightarrow NO VISTO CHE $A \rightarrow C$ e $A \rightarrow F$
NON HANNO UNA SUPERCHIAVE
A SX

3FN \rightarrow NO VISTO CHE IN $A \rightarrow C$ e $A \rightarrow F$
NON ABBIAMO NE LA SUPERCHIAVE
A SX NE UN SOTTOINSIEME DELLA CHIAVE
A DX.

ESERCIZIO 4 (2 punti)

a) (1 pt) Dato un DBMS relazionale, spiegare cosa si intende per proprietà ACID delle transazioni.

b) (1 pt) Sia dato un oggetto x , su cui opera un controllo della concorrenza basato su timestamp (mono-versione), con $WTM(x)=9$, $RTM(x)=12$. Dato il seguente schedule di operazioni:

$r_7(x) \rightarrow NO$
 $w_{10}(x) \rightarrow NO!$
 $w_{14}(x) \rightarrow OK$
 $r_{16}(x) \rightarrow OK$
 $w_{18}(x) \rightarrow OK$
 $r_{15}(x) \rightarrow NO$
 $r_{25}(x) \rightarrow OK$
 $w_{23}(x) \rightarrow NO$
 $r_{20}(x) \rightarrow OK$
 $w_{27}(x) \rightarrow OK$
 $r_{26}(x) \rightarrow NO$

Indicare quali operazioni sono consentite e quali abortite, ed il valore finale di $WTM(x)$ ed $RTM(x)$.

$WTM(x) = 27$
 $RTM(x) = 25$

2) ACID è un acronimo e intende le proprietà di:

Atomicità \rightarrow Regola da tutto o niente, durante un malfunzionamento del SW o vengono eseguite tutte le trans o annullate tutte

Consistenza \rightarrow non devono essere violati i vincoli di integrità

Isolamento \rightarrow ogni transazione è indipendente dalle altre

Durability \rightarrow nel caso di commit i dati devono essere salvati e non andare persi in caso di crash.