

WGUO 2016

NOME
COGNOME
MATRICOLA

ESERCIZIO 1 (12 punti)

Dato il seguente schema relazionale, che modella i dati di un sistema di gestione di campionati di basket.

GIOCATORE (Codice, Nome, Cognome, AnnoNascita) → costo
CONTRATTO (Id, CodiceGiocatore, NomeSquadra, Anno, Ingaggio)
SQUADRA (Nome, Città)

Con vincoli di integrità referenziale:

CONTRATTO.CodiceGiocatore → GIOCATORE.Codice
CONTRATTO.NomeSquadra → SQUADRA.Nome

a) (3 pt) Scrivere in SQL la query che determina codice, nome e cognome dei giocatori nati prima del 1988, che risultano SENZA contratto nell'anno 2013.

b) (2 pt) Scrivere in SQL la query che determina: numero delle squadre di Bologna che hanno avuto sotto contratto il giocatore "Andrea Rossi". → costo
[VINCOLO: Non è possibile usare il join tra tabelle nella clausola FROM]. gioc 1990

c) (3 pt) Scrivere in SQL la query che determina nome e città della squadra che ha avuto la spesa più alta in ingaggi nell'anno 2015, considerando SOLO gli ingaggi di giocatori nati dal 1990 in poi. [VINCOLO: Non è possibile utilizzare il costrutto HAVING]. NOME | ING

d) (2 pt) Scrivere in SQL il codice della tabella CONTRATTO, imponendo i seguenti vincoli: (i) Id è un alfanumerico di lunghezza pari esattamente a 10 caratteri, inizia sempre per "ID_" e termina sempre per "000"; (ii) Anno deve essere sempre definito; (iii) Un giocatore non può disporre di più di 3 contratti nello stesso anno.

d) (2 pt) Assumendo che GIOCATORE sia una Collezione in MONGO-DB, scrivere la query MONGO-DB che restituisce il codice (*) degli utenti di nome "Mario", cognome "Rossi", nati nel 1990.

(*) restituire solo i campi indicati

ESERCIZIO 2 (12 punti)

Si vuole progettare una base di dati per la gestione di una palestra. In particolare, si vogliono gestire i dati degli utenti in ingresso: ogni utente dispone di codice fiscale, nome, cognome, data di nascita, recapito email e recapito telefonico. Gli utenti possono appartenere a due categorie: utenti abbonati o utenti occasionali. Per i primi (utenti abbonati) si vuole tenere traccia della tipologia di abbonamento, data di inizio e data di fine. Per i secondi (utenti occasionali), si vuole tenere traccia della lista degli ingressi effettuati in palestra, con data e orario di ciascun ingresso. Inoltre, si vogliono memorizzare i dati relativi alle attrezzature di cui dispone la palestra: ogni attrezzo ha un codice (univoco), una marca, una descrizione, ed una o più foto associate. La base di dati gestisce inoltre le informazioni relative agli istruttori della palestra ed ai corsi da loro erogati. Ogni istruttore ha un nickname (univoco), nome, cognome, foto, e può essere temporaneo o permanente; gli istruttori permanenti dispongono anche di curriculum vitae (stringa di testo) e di qualifica (stringa di testo). Gli istruttori (sia temporanei sia permanenti) possono erogare corsi. Ogni corso dispone di un nome (es. fit boxe), una data di inizio, una data di fine, un giorno della settimana in cui viene svolto. Lo stesso corso può essere svolto da più istruttori. Di ogni corso, si vuole tenere traccia della lista degli utenti iscritti. Lo stesso utente può iscriversi a più corsi. Infine, si vuole tenere traccia delle schede di allenamento. Una scheda fa riferimento ad singolo utente abbonato, ed è redatta da un singolo istruttore permanente. Un utente abbonato dispone di una sola scheda a lui associata. Un istruttore permanente può redarre più schede di allenamento. La scheda dispone di una data di inizio, una data di fine, un livello di difficoltà (numero) e contiene l'elenco degli attrezzi da utilizzare; per ogni attrezzo della scheda si vuole tenere traccia del numero di ripetizioni e del carico (numero).

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

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

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

- Aggiungere un nuovo utente ed iscriverlo un corso esistente (Interattiva, 20 volte/mese).
- Visualizzare le informazioni (data inizio, data fine, difficoltà) relative a tutte le schede redatte da uno specifico istruttore (Interattiva, 1 volta/mese).
- Visualizzare le informazioni anagrafiche relative a tutti gli utenti della palestra (Interattiva, 2 volte/mese).

Tabella dei volumi: 30 utenti, 10 schede per istruttore

$$\alpha(\text{peso operazioni scrittura})=2, w_1(\text{peso operazioni interattive})=1$$

ESERCIZIO 3 (4 punti)

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

$$C \rightarrow AB \quad CA \rightarrow D \quad BF \rightarrow E \quad F \rightarrow D$$

a) (2pt) Indicare una superchiave ed una chiave dello schema. **Giustificare le risposte.**

b) (2pt) La relazione è in forma normale di Boyce e Codd (FNBC)? La relazione è in terza forma normale (3FN)? **Giustificare le risposte.**

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

↓
CHIÀVE VISTO CHE LA CHIUSURA
CONTIENE TUTTI GLI ATTRIBUTI
DELLA TABELLA R.
OVVIAMENTE È ANCHE SUPERCHIÀVE.

↓
CFA

↓
NO PERCHÉ NESSUNA
DELE DIPENDENTE È SUPERCHIÀVE

3FN → NO PERCHÉ NESSUNA DF
HA A DX UN SOTTOINSIEME DELLA
CHIÀVE

ESERCIZIO 4 (2 punti)

a) (1 pt) Dato un set di transazioni eseguite da un DBMS relazionale, spiegare (brevemente) la differenza tra schedule seriale e schedule serializzabile

b) (1 pt) Dato il seguente stato dei log: CK (CHECKPOINT) B (BACKUP) D (DELETE) C (COMMIT) U (UPDATE) I (INSERT)

CK(T₀) B(T₁) B(T₂) B(T₃) D(T₂, O₆, B₆) C(T₂) CK(T₀, T₁, T₃) C(T₀) U(T₃, O₃, B₃, A₃) B(T₄) B(T₅) I(T₄, O₄, A₄) C(T₄) guasto

Applicando l'algoritmo di ripresa a caldo, indicare la composizione degli insiemi UNDO e REDO (ossia, quali sono le transazioni di cui occorre fare l'undo e di quali il redo).

$$\text{UNDO} = \{T_1, T_3, O_6, B_6, O_3, B_3, A_3, T_5, O_4, A_4\}$$

$$\text{REDO} = \{T_4, T_0, T_2\}$$

A)

