Corso di Basi di Dati 1

Secondo Compitino

7 Gennaio 2003

Si consideri lo schema relazionale

SQUADRA(NomeS, Città, Sponsor, ColoriSociali, Allenatore)
GIOCATORE(NTessera, NomeS, Numero, Nome, Cognome, AnnoN, Ruolo)
PARTITA(IdPartita, Giornata, NomeSCasa, NomeSTrasf, GoalCasa, GoalTrasf)
GOAL(IdPartita, Minuto, NTessera, AutoGoal)

relativo alla base di dati di un campionato di calcio. Nella relazione PARTITA gli attributi NomeSCasa e NomeSTrasf memorizzano i nomi delle squadre che disputano la partita in casa e in trasferta, rispettivamente, e gli attributi GoalCasa e GoalTrasf rappresentano il numero di goal segnati da tali squadre. Nella relazione GOAL l'attributo NTessera memorizza il numero di tessera del giocatore che ha segnato il goal, mentre l'attributo AutoGoal é un attributo booleano che vale TRUE se il goal é stato un autogoal.

1. (Indicativamente, punti 4) Mostrare il comando SQL per definire la relazione GOAL, definendo i vincoli di chiave e chiave esterna per la relazione e i vincoli CHECK su attributo e su tupla che si ritengono opportuni. Commentare i comportamenti scelti in caso di violazioni dell'integrità referenziale.

2. Formulare in SQL le seguenti interrogazioni:

   (a) (indicativamente, punti 4) determinare l'allenatore della squadra che, in una sola partita, ha subito il maggior numero di goal;

   (b) (indicativamente, punti 5) per ogni giornata, determinare il portiere (della squadra) che in tale giornata subito più goal;

   (c) (indicativamente, punti 5) determinare, tra le squadre che hanno ottenuto in casa tutti i possibili risultati (sia vinto, che perso, che pareggiato), quella che dispone della rosa di giocatori più ampia e quella che dispone del maggior numero di attaccanti;

   (d) (indicativamente, punti 6) determinare le squadre che hanno vinto fuori casa in tutte le giornate in cui c'è stata almeno una vittoria fuori casa.

3. (Indicativamente, punti 5) Data la relazione

   RETI(NomeS, GoalFatti, GoalSubiti)

   che si suppone inizialmente vuota, definire una sequenza di comandi di aggiornamento opportuni per inserire in tale relazione le informazioni relative al numero di goal fatti e subiti da ogni squadra.

4. (Indicativamente, punti 3) Utilizzando la relazione RETI definire una query SQL che per ogni squadra mostri: nome squadra, goal fatti, goal subiti, differenza reti. Le squadre devono essere restituite ordinate in base alla differenza reti (miglior differenza reti per prima), in base ai goal fatti a parità di differenza reti, in ordine alfabetico a parità di differenza reti e goal fatti.

- CREATE TABLE R
  (spec_col$_1$, ..., spec_col$_n$)
  [, PRIMARY KEY (*Lista Nomi Colonne*)]
  [, UNIQUE (*Lista Nomi Colonne*)]
  [, FOREIGN KEY (*Lista Nomi Colonne*)
  REFERENCES R'
  [ON DELETE {NO ACTION | RESTRICT | CASCADE |
  SET NULL | SET DEFAULT}]
  [ON UPDATE {NO ACTION | RESTRICT | CASCADE |
  SET NULL | SET DEFAULT}]]
  [, FOREIGN KEY ...]
  [CHECK *Condizione*];
  ...
  [CHECK *Condizione*];
  );

  spec_col$_i$ = *Nome Colonna$_i$ Dominio$_i$* [DEFAULT *Valore Default*] [NOT NULL] [PRIMARY KEY] [UNIQUE]
  [CHECK *Condizione*], $\quad 1 \leq i \leq n$

  CREATE ASSERTION *Nome Vincolo*
  CHECK *Condizione*

- DROP TABLE R

- ALTER TABLE R ADD COLUMN spec_col;
  ALTER TABLE R ALTER COLUMN spec_col;
  ALTER TABLE R DROP COLUMN nome_colonna;

- SELECT $R_{i_1}.C_1, R_{i_2}.C_2, \ldots, R_{i_n}.C_n$
  FROM $R_1, R_2, \ldots, R_k$
  WHERE F;

  - C BETWEEN v_1 AND v_2
  - C IN $(v_1, \ldots, v_n)$ e C IN sq
  - C LIKE pattern, con pattern stringa di caratteri che può contenere i caratteri speciali % e_
  - EXISTS sq, NOT EXISTS sq
  - ANY sq, ALL sq
  - UNION, INTERSECT, EXCEPT tra query
  - funzioni di gruppo: MAX, MIN, SUM, AVG, COUNT - per le prime quattro è $f(nome_colonna)$, per COUNT ci sono le due varianti COUNT(*) e COUNT(DISTINCT nome_colonna)
  - l'ordine delle clausole (se presenti) è SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY

- INSERT INTO R [$(C_1, \ldots, C_n)$]
  { VALUES $(e_1, \ldots, e_n)$ | sq}

- DELETE R [alias] [WHERE F];

- UPDATE R [alias]
  SET C_1 = {e_1 | NULL }, ..., C_n = {e_n | NULL }
  [WHERE F];

- CREATE VIEW *Nome* [(*Lista Nomi Colonne*)] AS Q
  WITH [[LOCAL | CASCADED] CHECK OPTION];