

Compito di Basi di dati

2 febbraio 2015

Esercizio 1:

Sia dato il seguente schema relazionale:

MUSICISTA(CodiceIdentificativo, Nome, Cognome, CittàDiNascita, AnnoDiNascita);

HA_SUONATO_IN(Musicista, Sala, Anno);

SALA_DA_CONCERTO(Nome, Città, Capienza);

SI_TROVA_IN(Città, Nazione).

Si assuma che in ogni musicista sia individuato univocamente da un codice identificativo e sia caratterizzato da nome, cognome, città in cui è nato e anno di nascita. Si assuma, inoltre, che ogni sala da concerto sia identificata univocamente dal suo nome e sia caratterizzata dalla città in cui si trova e dalla capienza (numero di posti disponibili). Si assuma anche che vi possano essere più sale da concerto in una stessa città. Attraverso la relazione HA_SUONATO_IN si vuol tener traccia delle sale in cui un musicista ha suonato (una o più volte) un dato anno.

Definire preliminarmente le chiavi primarie, le eventuali altre chiavi candidate e, se ve ne sono, le chiavi esterne delle relazioni date. Successivamente, formulare opportune interrogazioni in algebra relazionale che permettano di determinare (senza usare l'operatore di divisione e usando solo se e quando necessario le funzioni aggregate):

- (i) i musicisti che non hanno mai suonato in una sala con una capienza maggiore di 500 posti o in una sala da concerto tedesca (in altri termini, nessun concerto in una sala con una capienza maggiore di 500 posti e nessun concerto in una sala da concerto tedesca);
- (ii) i musicisti che nel 2014 hanno suonato solo in sale da concerto nelle quali sono stati gli unici a suonare (nel 2014);
- (iii) i musicisti che hanno suonato in almeno una sala da concerto nella quale sono stati gli unici a suonare.

Esercizio 2:

Con riferimento all'Esercizio 1, formulare opportune interrogazioni in SQL che permettano di determinare quanto richiesto (senza usare l'operatore CONTAINS e usando solo se e quando necessario le funzioni aggregate).

Esercizio 3:

Si vuole progettare una base di dati di supporto all'organizzazione delle attività di una compagnia aerea italiana. La compagnia aerea offre un certo insieme di voli che possiedono un codice numerico che identifica la specifica tratta (ad esempio, Venezia-Roma), il tipo di aereo utilizzato, una data (21 gennaio 2015) e un orario di partenza (ore 16:00), una data (21 gennaio 2015) e un orario di arrivo (ore 18:30), un aeroporto di partenza e uno di destinazione. Di ogni aeroporto si memorizzano la città e la nazione in cui si trova e in numero di piste disponibili. Di ogni tipo di aereo si registrano la società costruttrice, il numero di passeggeri e la quantità di merci che possono essere trasportati. La compagnia offre sia voli

nazionali sia voli internazionali. I voli internazionali possono avere uno o più scali intermedi (tali scali vengono effettuati per ragioni di natura tecnica; non consentono ad alcun passeggero di salire o scendere). Dei voli passati interessa mantenere date e ore effettive di partenza e di arrivo (ad esempio, partenza il 21 gennaio 2015 alle ore 16:45, e arrivo il 21 gennaio 2015, alle ore 18:50); dei voli futuri sono di interesse il numero di posti prenotato e il numero di posti ancora disponibili.

Si definisca uno schema Entità-Relazioni che descriva il contenuto informativo del sistema, illustrando con chiarezza le eventuali assunzioni fatte. Lo schema dovrà essere completato con attributi ragionevoli per ciascuna entità (identificando le possibili chiavi) e relazione. Vanno specificati accuratamente i vincoli di cardinalità e partecipazione di ciascuna relazione.

Esercizio 4:

Si stabilisca se i seguenti schedule appartengono o meno a VSR, CSR, 2PL, 2PL stretto e TS.

1. $s_1 : r_1(x), r_3(z), w_2(x), r_4(y), w_4(z), r_2(x), w_1(x), r_3(y), w_4(y)$;
2. $s_2 : r_2(z), w_1(t), r_1(y), w_3(z), r_2(x), w_4(t), r_1(x), w_2(x), w_2(z), w_3(y), w_4(z)$;
3. $s_3 : r_4(y), w_1(x), r_1(y), r_2(t), w_3(t), r_2(y), w_2(x), w_4(y), r_1(z), w_4(x), r_4(t), w_3(z)$.

Esercizio 5:

Dato il seguente insieme di chiavi:

6, 10, 4, 17, 14, 3, 12, 8, 16, 2, 11, 7

mostrare il B -albero di ordine 4 ottenuto inserendo un elemento dopo l'altro nell'ordine dato (riportando la sequenza di alberi generata dal processo di inserimento).

Esercizio 6 (facoltativo):

Si consideri il seguente documento XML:

```
<A id="1">
  <B id="2">
    <E id="3" />
  </B>
  <C id="4">
    <E id="5" />
    <F id="6" />
  </C>
  <E id="7" />
</A>
```

Si determini il risultato di ciascuna delle seguenti interrogazioni XPath sul documento XML dato:

1. `/descendant::*[(not (child::*)) or (not (parent::*))]/attribute::id`
2. `//E[@id > 3]/../@id`