

Sistemi Informativi T
3 febbraio 2016

Tempo a disposizione: 2:30 ore

(Esame di SI L-A: esercizi 1) e 2), 1 ora; Esame di SI L-B: esercizi 3) e 4), 1:30 ore;)

La consegna deve essere eseguita mediante l'apposito applicativo Web, facendo l'upload dei file specificati sul sito <http://esamix.labx> (solo per l'es. 1 la consegna è su carta)

N.B. Per superare la prova di SI-T è necessario totalizzare almeno 3 punti negli esercizi 1 e 2

1) Algebra relazionale (3 punti totali):

Consegnare le risposte su un foglio di carta, intestato con matricola, nome e cognome

Date le seguenti relazioni, disponibili nello schema B16884 con dati fittizi di esempio:

```
PRODOTTI (CodP, Descrizione, Prezzo, Categoria);
LISTENOZZE (LID, CodP, NumQuote, PrezzoQuota);
          CodP REFERENCES PRODOTTI;
REGALI (LID, CodP, Invitato, NumQuote),
        (LID, CodP) REFERENCES LISTENOZZE;
--
-- LISTENOZZE.NumQuote: indica in quante quote (una o più)
-- il prezzo è diviso (Prezzo = PrezzoQuota*NumQuote)
-- REGALI.NumQuote (<= LISTENOZZE.NumQuote) è il numero di quote
-- di un dato prodotto che un invitato ha regalato
-- (la somma di REGALI.NumQuote non è maggiore del corrispondente
-- valore di LISTENOZZE.NumQuote)
-- Tutti i prezzi sono in formato DEC(6,2)
```

si scrivano in algebra relazionale le seguenti interrogazioni:

- 1.1) [1 p.]** I dati completi dei prodotti con prezzo minore di 250€ per cui esiste almeno una lista di nozze in cui il prodotto è stato completamente regalato da un singolo invitato
- 1.2) [2 p.]** Le liste di nozze (LID) in cui, per ogni categoria, compare almeno un prodotto con una sola quota

2) SQL (5 punti totali)

Consegnare il file SQL.txt

Con riferimento al DB dell'esercizio 1, si scrivano in SQL le seguenti interrogazioni:

- 2.1) [2 p.]** Per ogni lista di nozze e per ogni prodotto relativo con almeno due quote e regalato da almeno un invitato, il rapporto (con due cifre decimali) tra le quote sinora pagate e le quote totali (es. prodotto presente con 3 quote di cui 2 già regalate: $2/3 = 0,66$)
- 2.2) [3 p.]** Per ogni fascia di prezzo (0-499 €, 500-999 €, ecc.), il numero di liste di nozze in cui la somma sinora pagata dagli invitati per i regali cade in quella fascia

3) Progettazione concettuale (6 punti)

Consegnare il file ER.lun

“1000 Patrimoni da Salvare” (1kPdS) è un’associazione che si occupa della tutela e salvaguardia del patrimonio artistico e naturalistico nazionale. 1kPdS è organizzata in sezioni, ognuna con uno o più soci. Ogni socio mette a disposizione le proprie competenze, descritte secondo il sistema di classificazione della 1kPdS, che prevede sia una tipologia di classificazione (ad es. fotografia) che un livello (ad es. esperto).

Periodicamente ogni sezione organizza delle visite di ispezione presso un elemento tutelato da quella sezione (ogni elemento è a carico di una sola sezione). I vari elementi hanno una tipologia (monumento, sito naturalistico, ecc.), un nome, un codice univoco e una descrizione. Ogni visita viene effettuata da uno o più soci della sezione e produce della documentazione specifica, che viene archiviata specificandone nome (univoco per quella visita) e contenuto.

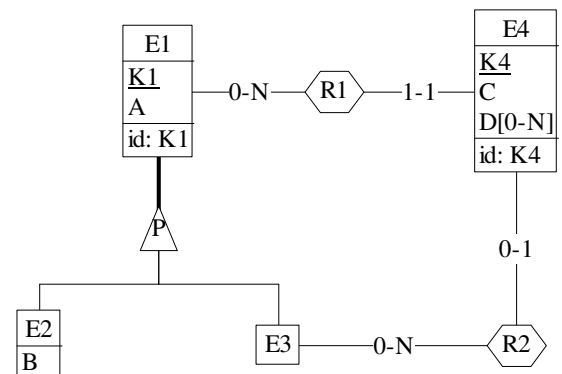
Annualmente la 1kPdS rende disponibile un “rapporto di salute” di tutti gli elementi tutelati, in cui per ogni elemento viene fornito un quadro riassuntivo della situazione, corredato da alcuni elementi della documentazione prodotta durante le relative visite.

4) Progettazione logica (6 punti totali)

Consegnare i file SCHEMI.txt e TRIGGER.txt

Dato lo schema concettuale in figura e considerando che:

- tutti gli attributi sono di tipo INT;
- le entità E1, E2 ed E3 vengono tradotte assieme;
- le associazioni R1 e R2 non vengono tradotte separatamente;
- un’istanza di E4 che partecipa a R2 non può aver nessun valore di D maggiore di 20;



4.1) [3 p.] Si progettino gli opportuni schemi relazionali e si definiscano tali schemi in DB2 (sul database SIT_STUD) mediante un file di script denominato **SCHEMI.txt**

4.2) [3 p.] Per i vincoli non esprimibili a livello di schema si predispongano opportuni **trigger** che evitino inserimenti di singole tuple non corrette, definiti in un file **TRIGGER.txt** e usando se necessario il simbolo '@' per terminare gli statement SQL (altrimenti ';')

IMPORTANTE:

- I file **NON** devono includere istruzioni di (dis)connessione al DB
- Per il punto 4.2), se necessario, si specifichino usando commenti SQL eventuali inserimenti di tipo transazionale (ossia, più INSERT nella stessa transazione)
- Si prega di attenersi scrupolosamente alle istruzioni relative ai nomi dei file (maiuscole incluse). **Il mancato rispetto delle istruzioni potrà comportare penalizzazioni di punteggio**