

Sistemi Informativi T
16 gennaio 2013

Tempo a disposizione: 2:30 ore

SI L-A: (solo esercizi 1) e 2)) 1 ora;

SI L-B: (solo 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 (ProdID, Categoria, Prezzo);
RECLAMI (RID, ProdID, Data, Cliente, Motivo),
ProdID REFERENCES PRODOTTI;
ESITI (RID, DataSoluzione, Esito, Rimborso*),
RID REFERENCES RECLAMI;
-- DataSoluzione = la data in cui si e' deciso come trattare il reclamo
-- Esito = descrive se e come il reclamo è stato accolto
-- Se Esito = 'RIMBORSO', allora l'attributo Rimborso riporta l'importo
--   rimborsato al cliente (minore o uguale del prezzo del prodotto),
--   altrimenti Rimborso è NULL
-- Prezzo e Rimborso sono di tipo DEC(8,2)
```

si scrivano in algebra relazionale le seguenti interrogazioni:

- 1.1) [1 p.]** I dati dei reclami che sono stati risolti in meno di 30 giorni e che hanno dato luogo a un rimborso almeno pari all'80% del prezzo del prodotto
 - 1.2) [2 p.]** I dati dei prodotti di categoria 'elettrodomestici' di prezzo superiore ai 100 € per cui esiste un reclamo senza esito
-

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.]** I dati dei prodotti di categoria 'elettrodomestici' di prezzo superiore ai 100 € per cui esiste un reclamo senza esito
- 2.2) [3 p.]** La categoria di prodotti con la massima percentuale di reclami rimborsati, considerando solo i reclami con esito definito

NB: L'espressione `DAYS(Data1) - DAYS(Data2)` restituisce il numero di giorni trascorsi tra due date (con `Data2 <= Data1`)

3) Progettazione concettuale (6 punti)

Consegnare il file ER.lun

Il sito FourWheels (4W) dà la possibilità di accedere a tutte le informazioni sul mercato delle auto usate. Oltre alle caratteristiche di base (nome, cilindrata, alimentazione) dei **modelli**, per ogni specifica **auto** vengono forniti i dati di utilizzo (anno immatricolazione, km percorsi, prezzo richiesto ed eventuali note) e quelli del **rivenditore** (nome, indirizzo, uno o più telefoni, email ed eventuale sito web). 4W consente a **chi si registra** (tramite un indirizzo email, che funge da username, e una password) di prenotare **test di prova** delle auto di proprio interesse, concordando per ogni test data e ora. Al termine di un test, il venditore lo comunica a 4W e l'utente, se interessato, può così scrivere sul sito le impressioni avute dal test e quindi ricevere un codice utile per ottenere uno sconto sull'acquisto dell'auto.

Per i clienti che prenotano un test senza né disdirlo (che comporta semplicemente la cancellazione della relativa prenotazione) né effettuarlo, 4W conteggia 1 punto negativo. Totalizzati 3 punti l'utenza viene disabilitata e non si possono più creare registrazioni con la stessa email (pertanto solo gli utenti attivi, ovvero non disabilitati, possono avere dei test ancora da eseguire).

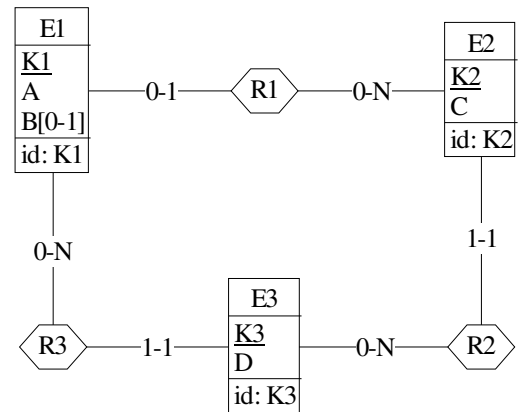
4) Progettazione logica (6 punti totali)

Consegnare i file SCHEMI.txt e TRIGGER.txt

Dato lo schema concettuale in figura e considerando che:

- tutti gli attributi sono di tipo INT;
- nessuna associazione viene tradotta separatamente;
- un'istanza di E1 non è mai associata, tramite le associazioni R1, R2 e R3, a un'istanza di E1 con B nullo;

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 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**