

Tempo a disposizione: 2:30 ore

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

Corso 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:

```
MEDICI (MID, Nome, DataNascita, DataAssunzione, Reparto) ,
    Reparto REFERENCES REPARTI;
REPARTI (Nome, IDDirettore*) ,
    IDDirettore REFERENCES MEDICI;
RICOVERI (RID, NomePaziente, Motivo, Reparto, Da, A*) ,
    Reparto REFERENCES REPARTI;
-- A ha valore nullo per i pazienti ancora ricoverati
-- Ogni direttore lavora nel reparto che dirige
-- Un reparto può anche non avere un direttore
```

si scrivano in algebra relazionale le seguenti interrogazioni:

- 1.1) [1 p.]** I pazienti che sono stati ricoverati più di 30 giorni nel reparto diretto dal dottor Rossi (quello nato nel 1955) e il cui ricovero è terminato
- 1.2) [2 p.]** I nomi dei reparti che nel 2014 non hanno assunto nessun medico con meno di 35 anni (alla data dell'assunzione)
-

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 nomi dei reparti che nel 2014 non hanno assunto nessun medico con meno di 35 anni (alla data dell'assunzione)
- 2.2) [3 p.]** Per ogni fascia di età (20-29, 30-39, ecc.), il reparto con il maggior numero di medici, escludendo dal conteggio i direttori

NB:

- L'espressione YEAR(Data) restituisce un anno in formato intero (es.: YEAR('21/01/2015') = 2015)
- L'espressione YEAR(Data1 - Data2) restituisce il numero di anni trascorsi da Data2 a Data1 (con Data2 <= Data1) (es.: YEAR('21/01/2015' - '23/02/2014') = 0, YEAR('21/01/2015' - '20/01/2013') = 2)
- L'espressione DAYS(Data1) - DAYS(Data2) restituisce il numero di giorni trascorsi tra due date (con Data2 <= Data1) (es.: DAYS('21/01/2015') - DAYS('19/12/2014') = 33)

3) Progettazione concettuale (6 punti)

Consegnare il file ER.lun

Il consorzio di agricoltori eCampi riunisce diverse aziende agricole.

Ogni azienda gestisce un proprio magazzino, in cui mantiene la frutta e verdura disponibile. Per ogni prodotto viene registrato il numero di cassette prodotte e quante ve ne sono ancora in magazzino, distinguendo sulla base delle diverse date di produzione (ad es. l'azienda Gregoretti ha prodotto 500 cassette di bietole il 12/12/2014 e in magazzino ce ne sono ancora 250, per la produzione del 10/01/2015 i numeri sono 200 e 150, rispettivamente). La dimensione di una cassetta (peso medio) è fissa per un dato prodotto (ad es. il peso medio di una cassetta di bietole è di 30 kg).

Per ottimizzare le produzioni, la gestione dei magazzini e la rete di vendita, le aziende della eCampi ricorrono spesso a movimenti di prodotti da un magazzino all'altro. Ogni movimento riguarda di norma diversi prodotti, e per esso vengono registrati data e ora di partenza e di arrivo (le seconde solo a consegna avvenuta), e per ogni prodotto inviato il numero di cassette (con relative date di produzione).

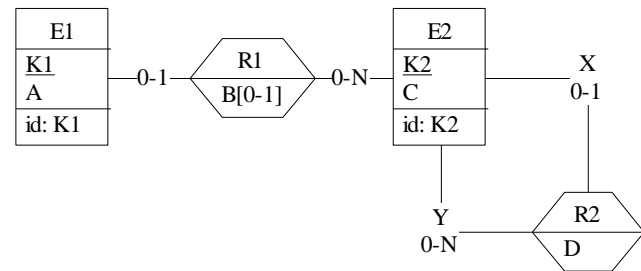
Il sistema informativo di eCampi mantiene solo la situazione attuale delle giacenze nei magazzini, ma tiene traccia di tutte le produzioni e dei movimenti eseguiti, senza limiti di tempo.

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 associazioni R1 e R2 non vengono tradotte separatamente;
- il valore di B è sempre definito se la relativa istanza di E1 è associata a un'istanza di E2 che partecipa all'associazione R2 con ruolo X;
- Per ogni istanza di E2, il valore di C è almeno pari alla somma dei valori di A delle istanze di E1 a essa associate;



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**