

Informatica documentale
Laurea in Scienze della Comunicazione
Prova scritta del 28 gennaio 2013

Cognome e nome: _____ Matricola: _____

Parte prima	Domanda 1	Domanda 2	Domanda 3	Totale

Istruzioni:

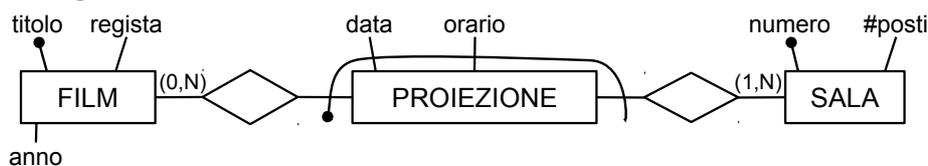
- È vietato portare all'esame libri, eserciziari, appunti e dispense. Chiunque venga trovato in possesso di documentazione relativa al corso – anche se non attinente alle domande proposte – vedrà annullata la propria prova.
- Scrivere solo sui fogli distribuiti, cancellando le parti di brutta con un tratto di penna. Non separare questi fogli.
- Tempo a disposizione: 1 ora e 45 minuti.

Parte prima

- 1) In al più 5 righe dire cosa significano le sigle DDL e DML nell'ambito dei linguaggi per le basi di dati?
- 2) In al più 10 righe descrivere a cosa servono e quali sono le azioni compensative (o politiche di reazione) alla violazione di un vincolo d'integrità referenziale.
- 3) In al più 10 righe spiegare cosa si intende per indipendenza dei dati in un DBMS.

Parte seconda

- 1) Considerare il seguente schema ER.



- a) In al più 5 righe, spiegare che tipo di entità è PROIEZIONE e cosa significa la linea che la circonda.
- b) Solo sulla base dell'ER sopra (cioè senza ulteriori vincoli e/o informazioni) è possibile per un film essere proiettato più volte (in una stessa sala o in sale diverse)? Spiegare brevemente (max 3 righe) perché.
- c) Nell'ER aggiungere alle relazioni le cardinalità mancanti.
- d) All'entità PROIEZIONE aggiungere un attributo facoltativo "biglietti_venduti" che vuole rappresentare il numero di biglietti venduti per una proiezione.
- e) Tradurre lo schema ER in uno schema relazionale, indicando in quest'ultimo eventuali chiavi, vincoli di non nullità e vincoli di integrità referenziale.

2) Considerare il seguente schema relazionale che rappresenta la distribuzione delle pubblicazioni nelle varie edicole.

- FIERA (titolo, data_inizio, data_fine)
- ESPOSITORE (nome, cittaSede)
- POSTAZIONE (espositore, fiera, padiglione, stand)
con vincoli di integrità referenziale tra l'attributo espositore di POSTAZIONE con la relazione ESPOSITORE e tra l'attributo fiera di POSTAZIONE con la relazione FIERA.

e la seguente istanza.

FIERA		
<u>titolo</u>	<u>data_inizio</u>	<u>data_fine</u>
Sport Expo	16/03/2013	18/03/2013
Vinitaly	07/04/2013	10/04/2013
Mineral Show	03/05/2013	05/05/2013

ESPOSITORE	
<u>nome</u>	<u>cittaSede</u>
Masi	Verona
Allegrini	Verona
Salewa	Bolzano
Mikado	Treviso
Arena	Macerata

POSTAZIONE			
<u>espositore</u>	<u>fiera</u>	<u>padiglione</u>	<u>stand</u>
Masi	Vinitaly	2	101
Allegrini	Vinitaly	2	57
Salewa	Sport Expo	1	12
Mikado	Sport Expo	2	38
Arena	Sport Expo	1	46

- Dare i comandi SQL per creare le tre tabelle sopra (incluse eventuali chiavi, vincoli di non nullità, vincoli di unicità e vincoli di integrità referenziale). Tenere conto che deve essere evitato di cancellare un espositore o una fiera a cui fa riferimento almeno una tupla di POSTAZIONE, e che in caso di modifica del titolo di una fiera, questa modifica si deve riflettere in automatico anche in POSTAZIONE.
- Dare il comando SQL per inserire in FIERA la fiera "ArtVerona" dal 10/10/2013 al 14/10/2013.
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT fiera, count(*) as espositori
FROM POSTAZIONE
GROUP BY fiera
```
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT espositore, fiera, stand
FROM POSTAZIONE
WHERE padiglione=1
```
- Dire se il seguente comando è corretto, e in tal caso scrivere la tabella risultante dalla sua esecuzione nell'istanza sopra, o no, e in tal caso spiegare (max 3 righe) perché non lo è.

```
INSERT INTO POSTAZIONE VALUES ('Masi', 'Vinitaly', 2, 102)
```

3) Considerare la base di dati definita tramite i seguenti comandi SQL.

```
CREATE TABLE paziente (  
    cf char(16) PRIMARY KEY,  
    nome varchar(32) NOT NULL,  
    cognome varchar(32) NOT NULL,  
    eta integer  
);
```

```
CREATE TABLE stanza (  
    codice integer PRIMARY KEY,  
    edificio varchar(32) NOT NULL,  
    piano integer NOT NULL,  
    numero integer NOT NULL,  
    reparto varchar(32),  
    UNIQUE (edificio, piano, numero)  
);
```

```
CREATE TABLE visita (  
    paziente char(16) NOT NULL REFERENCES paziente(cf) ON UPDATE CASCADE,  
    stanza integer REFERENCES stanza(codice) ON UPDATE CASCADE,  
    data date NOT NULL,  
    ora time NOT NULL,  
    tipologia varchar(32) NOT NULL,  
    motivo varchar(128),  
    PRIMARY KEY (stanza, data, ora),  
    UNIQUE (paziente, data, ora)  
);
```

La base di dati rappresenta la programmazione delle visite in un ospedale. Ogni visita è allocata in una certa stanza ad una precisa data e ora. Ad ogni visita è anche associato il paziente che la deve sostenere.

a) Cosa significa - stanza integer REFERENCES stanza(codice) - utilizzato nella definizione della tabella visita?

Formulare le interrogazioni SQL che permettano di determinare:

- b) tutte le informazioni riguardanti i pazienti con più di 60 anni;
- c) nome e cognome dei pazienti che devono subire visite la cui tipologia è **oculistica**;
- d) per ogni data, il numero di visite che vi sono programmate;
- e) nome e cognome del paziente che deve subire una visita, insieme alla data della visita e alla stanza (edificio, piano e numero) in cui si tiene.