

Informatica documentale
Laurea in Scienze della Comunicazione
Prova scritta del 25 giugno 2012

Cognome e nome: _____ Matricola: _____

Parte prima	Domanda 1	Domanda 2	Domanda 3	Totale

Istruzioni:

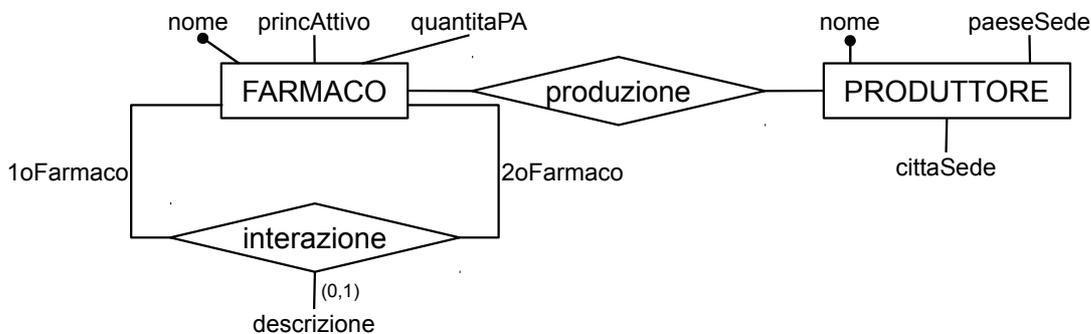
- È vietato portare all'esame libri, eserciziari, appunti e dispense. Chiunque venga trovato in possesso di documentazione relativa al corso – anche se non attinente alle domande proposte – vedrà annullata la propria prova.
- Scrivere solo sui fogli distribuiti, cancellando le parti di brutta con un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

Parte prima

- 1) In al più 10 righe dire cosa significa che i DBMS devono poter gestire basi di dati *grandi, persistenti e condivise*?
- 2) In al più 5 righe dire cosa significano le sigle DDL e DML nell'ambito dei linguaggi per le basi di dati?
- 3) Quali sono le due possibili classificazioni delle generalizzazioni nel modello ER?
- 4) Come si rappresenta nel modello relazionale, e poi nello standard SQL, l'informazione incompleta o la mancanza di informazione?
- 5) Dare la rappresentazione grafica nel modello ER di un'entità debole (o, in altre parole, di un identificatore esterno).

Parte seconda

- 1) Considerare il seguente schema ER.



- a) Aggiungere le cardinalità delle relazioni.
- b) Spiegare brevemente (max 3 righe) che tipo di relazione è *interazione* e cosa rappresentano *1oFarmaco* e *2oFarmaco*.
- c) Solo sulla base dell'ER sopra (cioè senza ulteriori vincoli e/o informazioni) è possibile avere in *interazione* un farmaco in relazione con sé stesso? Spiegare brevemente (max 3 righe) perché.
- d) E' possibile avere contemporaneamente nell'entità FARMACO le seguenti due istanze? (Tachipirina, paracetamolo, 50mg) (Paracet50, paracetamolo, 50mg) Spiegare brevemente (max 3 righe) perché.
- 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 il palinsesto dei canali televisivi.

- CANALE (numero, nome)
- PROGRAMMA (nome, tipo*, genere*, classificazione*, durata)
- MESSAINONDA (programma, canale, data, orainizio, orafine, numstacchipubblicitari*)
con vincoli di integrità referenziale tra l'attributo programma di MESSAINONDA con la relazione PROGRAMMA e tra l'attributo canale di MESSAINONDA con la relazione CANALE.

e la seguente istanza.

PROGRAMMA				
nome	tipo	genere	classificazione	durata
Blob	NULL	NULL	T	00:20
TG3	informazione	NULL	T	00:30
Cloverfield	film	azione	VM12	01:40

CANALE	
numero	nome
101	RAI1
102	RAI2
103	RAI3

MESSAINONDA					
programma	canale	data	orainizio	orafine	numstacchipubblicitari
TG3	103	30-06-2012	13:00	13:30	0
Cloverfield	102	30-06-2012	21:10	23:00	3
TG3	103	30-06-2012	07:00	07:30	0

- 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 l'attributo nome della relazione CANALE non può contenere ripetizioni e che deve essere evitato di cancellare un programma o un canale a cui almeno una messa in onda fa riferimento.
- Dare il comando SQL per inserire in MESSAINONDA la trasmissione su RAI3 dalle 23:00 alle 23:30 del giorno 30-06-2012 del programma Blob.
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT c.nome as nome, count(programma) as programmi
FROM CANALE AS c LEFT JOIN MESSAINONDA AS mio ON (c.numero=mio.canale)
GROUP BY c.nome
```
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT canale, data, orainizio
FROM MESSAINONDA
WHERE programma = 'Cloverfield'
```
- 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 PROGRAMMA VALUES ('TG3 Salute', 'informazione', NULL, '00:25')
```

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,  
    piano integer NOT NULL,  
    numero integer NOT NULL,  
    reparto varchar(32),  
    UNIQUE (piano, numero)  
);
```

```
CREATE TABLE visita (  
    paziente char(16) REFERENCES paziente(cf) ON DELETE NO ACTION ON UPDATE CASCADE,  
    stanza integer REFERENCES stanza(codice) ON DELETE NO ACTION 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 ON UPDATE CASCADE utilizzato nella definizione del campo paziente della tabella visita?

Formulare le interrogazioni SQL che permettano di determinare:

- b) nome e cognome dei pazienti la cui età è minore o uguale a 14;
- c) nome e cognome dei pazienti con una visita programmata per il giorno 30-06-2012;
- d) per ogni data, il numero di visite in tale data;
- e) l'età media dei pazienti con una visita la cui tipologia è “oncologica”;

SOLUZIONI

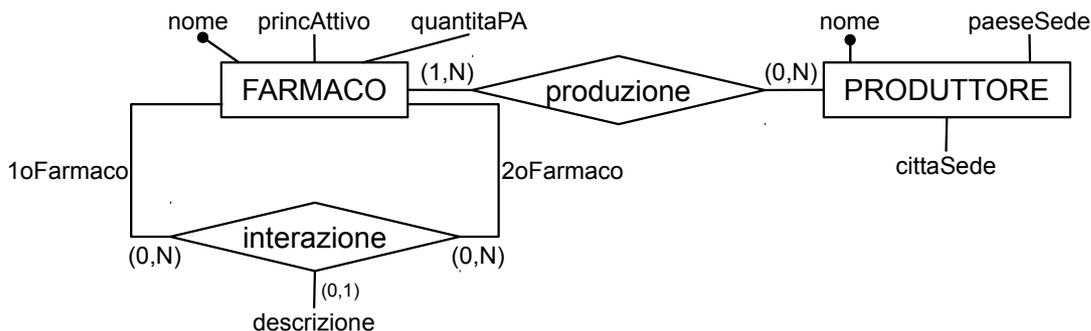
Parte prima

- 1) *Grandi*: le basi di dati possono avere dimensioni (molto) maggiori della memoria centrale dei sistemi di calcolo utilizzati il limite deve essere solo quello fisico dei dispositivi.
Persistenti: le basi di dati hanno un tempo di vita indipendente dalle singole esecuzioni dei programmi che le utilizzano.
Condivise: una base di dati è condivisa da più applicazioni ed utenti che vi possono accedere anche contemporaneamente e concorrentemente.
- 2) DDL: Data Definition Language, linguaggio per la definizione di schemi e altre operazioni generali
DML: Data Manipulation Language, linguaggio per l'interrogazione e l'aggiornamento (inserimento, cancellazione, modifica) di istanze di basi di dati
- 3) Totale-Parziale e Esclusiva-Sovrapposta
- 4) L'informazione incompleta o la sua mancanza si rappresenta con il valore NULL.
- 5)



Parte seconda

- 1)
 - a) Aggiungere le cardinalità delle relazioni.



- b) **interazione** è una relazione ricorsiva e **1oFarmaco** e **2oFarmaco** sono i ruoli svolti dalle due diverse istanze di **FARMACO**.
- c) Si è possibile, perché **interazione** è identificata dai nomi di entrambi i farmaci che compongono l'interazione e non vi è nessun vincolo nell'ER sul fatto che questi due farmaci debbano essere diversi.
- d) Si è possibile, perché l'unico vincolo in **FARMACO** è il vincolo di chiave e le due tuple lo rispettano avendo chiavi (nomi) diversi.
- e) **FARMACO**(nome, princAttivo, quantitaPA)
PRODUTTORE(nome, paeseSede, cittaSede)
PRODUZIONE(nomeFarmaco, nomeProduttore)
con vincoli di integrità referenziale tra l'attributo **nomeFarmaco** di **PRODUZIONE** con la relazione **FARMACO** e tra l'attributo **nomeProduttore** di **PRODUZIONE** con la relazione **PRODUTTORE**.
INTERAZIONE(1oFarmaco, 2oFarmaco, descrizione*)
con vincoli di integrità referenziale tra gli attributi **1oFarmaco** e **2oFarmaco** di **INTERAZIONE** con la relazione **FARMACO**.

- 2)
- a) CREATE TABLE canale (
 - numero integer PRIMARY KEY,
 - nome varchar(32) NOT NULL UNIQUE
);

 CREATE TABLE programma (
 - nome varchar(32) PRIMARY KEY,
 - tipo varchar(32),
 - genere varchar(32),
 - classificazione varchar(32),
 - durata time NOT NULL
);

 CREATE TABLE messainonda (
 - programma varchar(32) REFERENCES programma(nome) ON DELETE NO ACTION,
 - canale integer REFERENCES canale(numero) ON DELETE NO ACTION,
 - data date,
 - orainizio time,
 - orafine time NOT NULL,
 - numstacchipubblicitari integer,
 - PRIMARY KEY (programma, canale, data, orainizio)
);
 - b) INSERT INTO messainonda (programma, canale, data, orainizio, orafine) VALUES ('Blob', 103, '30-06-2012', '23:00', '23:30');

 oppure

 INSERT INTO messainonda VALUES ('Blob', 103, '30-06-2012', '23:00', '23:30', NULL);
 - c)

nome	programmi
RAI1	0
RAI2	1
RAI3	2
 - d)

canale	data	orainizio
102	30-06-2012	21:10
 - e) Non è corretto perché, non avendo specificato la lista degli attributi di cui si specificano i valori, è obbligatorio specificare un valore per ogni attributo della relazione PROGRAMMA, mentre nel comando dato ne manca uno.
- 3)
- a) ON UPDATE CASCADE significa che ogni modifica ad un valore del campo cf della tabella paziente viene “a cascata” e automaticamente eseguita anche sul campo paziente di tutte le tuple della tabella visita che fanno riferimento al valore modificato.
 - b) SELECT nome, cognome

 FROM paziente

 WHERE eta <= 14
 - c) SELECT nome, cognome

 FROM paziente JOIN visita ON cf = paziente

 WHERE data = '30-06-2012'
 - d) SELECT data, count(*)

 FROM visita

 GROUP BY data
 - e) SELECT avg(eta)

 FROM paziente JOIN visita ON cf = paziente

 WHERE tipologia = 'oncologica'