

Informatica documentale
Laurea in Scienze della Comunicazione
Prova scritta del 18 luglio 2012

Cognome e nome: _____ Matricola: _____

Parte prima	Domanda 1	Domanda 2	Domanda 3	Totale

Istruzioni:

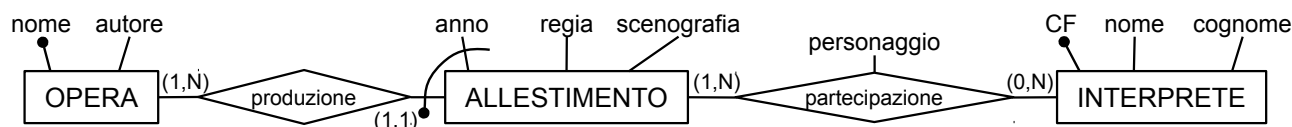
- È vietato portare all'esame libri, eserciziari, appunti e dispense. Chiunque venga trovato in possesso di documentazione relativa al corso – anche se non attinente alle domande proposte – vedrà annullata la propria prova.
- Scrivere solo sui fogli distribuiti, cancellando le parti di brutta con un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

Parte prima

- 1) In al più 5 righe descrivere la differenza nei sistemi informativi tra *informazione* e *dato*.
- 2) In al più 5 righe definire cosa sono *schema* ed *istanza* di una base di dati.
- 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 3 righe, spiegare che tipo di entità è ALLESTIMENTO.
- b) Solo sulla base dell'ER sopra (cioè senza ulteriori vincoli e/o informazioni) è possibile per un interprete interpretare più personaggi in uno stesso allestimento? Spiegare brevemente (max 3 righe) perché.
- c) Solo sulla base dell'ER sopra (cioè senza ulteriori vincoli e/o informazioni) è possibile che uno stesso personaggio sia interpretato da più interpreti in uno stesso allestimento? Spiegare brevemente (max 3 righe) perché.
- d) In al più 3 righe spiegare cosa significa (0,N) scritto tra INTERPRETE e PARTECIPAZIONE.
- 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.

- PUBBLICAZIONE (nome, periodicità)
- EDICOLA (codice, proprietario*, indirizzo, città)
- DISTRIBUZIONE (pubblicazione, edicola, data, numeroCopie)
con vincoli di integrità referenziale tra l'attributo pubblicazione di DISTRIBUZIONE con la relazione PUBBLICAZIONE e tra l'attributo edicola di DISTRIBUZIONE con la relazione EDICOLA.

e la seguente istanza.

EDICOLA			
<u>codice</u>	proprietario	indirizzo	città
V10	Carlo Rossi	Via Roma, 17	Verona
T31	NULL	Via Apollo, 42	Treviso
V12	Giada Bianchi	Piazza 25 aprile	Verona

PUBBLICAZIONE	
<u>nome</u>	periodicità
Internazionale	settimanale
L'Arena	quotidiano
Bresciaoggi	quotidiano

DISTRIBUZIONE			
<u>pubblicazione</u>	<u>edicola</u>	<u>data</u>	numeroCopie
Internazionale	V10	18-07-2012	100
L'Arena	V10	18-07-2012	200
Bresciaoggi	V12	18-07-2012	40
Bresciaoggi	V12	19-07-2012	50
L'Arena	V10	19-07-2012	150

- 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 una pubblicazione o un'edicola a cui fa riferimento almeno una tupla di DISTRIBUZIONE, e che in caso di modifica del nome di una pubblicazione, questa modifica si deve riflettere in automatico anche in distribuzione.
- Dare il comando SQL per inserire in DISTRIBUZIONE l'invio all'edicola V12 di 70 copie della pubblicazione "Internazionale" in data 18-07-2012.
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT pubblicazione, edicola, sum(numeroCopie) as copie
FROM DISTRIBUZIONE
GROUP BY pubblicazione, edicola
```
- Cosa restituisce la seguente interrogazione? (scrivere la tabella risultante)

```
SELECT *
FROM EDICOLA
WHERE proprietario IS NULL
```
- 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 DISTRIBUZIONE VALUES ('Bresciaoggi', 'V10', NULL, 75)
```

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 UNIQUE(edificio, piano, numero) utilizzato nella definizione della stanza?

Formulare le interrogazioni SQL che permettano di determinare:

- b) tutte le informazioni riguardanti le visite il cui motivo è sconosciuto;
- c) la data delle visite, con nome e cognome dei pazienti che le devono subire, programmate nella stanza con codice T3;
- d) per ogni reparto, il numero di stanze che vi sono presenti;
- e) edificio, piano e numero di tutte le stanze e, per ciascuna, il numero di visite che vi sono state eseguite (se una stanza non è mai stata utilizzata per una visita nel risultato dovrà comparire con 0 visite).