

Database Systems, CSCI 4380-01 Exam #2
Thursday March 24, 2010 at 2 pm

1 (12)	2 (12)	3 (14)	4 (12)	5 (12)	6 (12)	7 (14)	8 (12)	TOTAL

Note. The exam is open book and open notes. Use your own book and notes only, sharing is not allowed. Electronic gadgets are **NOT** allowed during the exam. Write your answers clearly, legibly and explain your reasoning as much as you can. If I cannot read or understand your answers, you will not get points.

Answer questions 1-7 in this exam using the data model in the appendix. Use SQL for questions 1-6.

Question 1 (12 points). Find all the books authored by 'Barack Obama' on topics 'Politics' or 'Biography'. Return the id, title of the books.

Question 2 (12 points). For each book author, return their id and name, the total number of books they wrote, the average price of these books and the total number of copies for these books (see attribute `copiesPublished`). (Note. only consider version 1 of each book.)

Question 3 (14 points). For each book author, find the latest book(s) they wrote. Return the id of the person and the id of the book(s). You may end up returning more than one book if the person wrote multiple books in that year.

Question 4 (12 points). Update the people table by setting the numPublishers for each person to be the total number of different publishers this person has published books with (use attribute books.publisher to find the publisher of a book that the person has authored).

Question 5 (12 points). Write an assertion for the following: If the versionNo for a book is greater than 1, then there must exist a previous version for that book in the database.

Question 6 (12 points). Find people who never authored and edited the same book, and did not author two books on the same topic. Return the id, name of the people.

Question 7 (14 points). Suppose you want to expand on the given data model to implement a social networking functionality. You will store information about users who comment on books, post book lists and mark books as well as authors as their favorites. Either draw an ER diagram for the expansion of the data model, or list the new tables and/or changes to the existing model. Make sure you show the connections of the new information to the existing tables.

In your expansion, you will store relevant information about users (users are not stored in the people table but in a new table). Users can add books or people to their favorites. Users can comment on books with a text and rating between 1 and 5. Users can also create book lists. Each list has a name and a number of books with a position in the list and an optional text describing it (Book 1: X, 'A great book', Book 2: Y, 'An introductory text', etc.) Mark clearly the primary keys and foreign keys.

Question 8 (12 points). You are given the following table definitions and table instance. Write down the result of executing the following insert/update/delete operations on the tables below. Make sure you list the contents of each table after each operation. Assume each operation executes using the given tables, not consecutively. For each query, write a sentence explaining why the final contents of the tables are as given.

```
CREATE TABLE r1 (
  a    INT PRIMARY KEY
  , b  INT ) ;
```

```
CREATE TABLE r2 (
  c    INT PRIMARY KEY
  , d  INT ) ;
```

```
CREATE TABLE r3 (
  e    INT PRIMARY KEY
  , f  INT NOT NULL
  , g  INT
  , FOREIGN KEY (f) REFERENCES r1(a)
    ON DELETE CASCADE ON UPDATE CASCADE
  , FOREIGN KEY (g) REFERENCES r2(c)
    ON DELETE SET NULL ON UPDATE SET NULL ) ;
```

```
CREATE TABLE r4 (
  h    INT
  , i  INT
  , PRIMARY KEY (h,i)
  , FOREIGN KEY (i) REFERENCES r3(e)
    ON DELETE CASCADE
    ON UPDATE SET NULL ) ;
```

r1	
a	b
a1	b1
a2	b1
a3	b2

r2	
c	d
c1	d1
c2	d2
c3	d3

r3		
e	f	g
e1	a1	c1
e2	a3	c1
e3	a1	c2
e4	a2	c2

r4	
h	i
h1	e1
h2	e2
h2	e2
h3	e4

a. DELETE FROM r1 WHERE b = 'b1' ;

c. UPDATE r1 SET a = 'a4' WHERE b = 'b1' ;

r1	
a	b

r2	
c	d

r3		
e	f	g

r4	
h	i

r1	
a	b

r2	
c	d

r3		
e	f	g

r4	
h	i

b. update r2 SET d = 'd4' WHERE c = 'c2';

d. DELETE FROM r3 WHERE e = 'e4' ;

r1	
a	b

r2	
c	d

r3		
e	f	g

r4	
h	i

r1	
a	b

r2	
c	d

r3		
e	f	g

r4	
h	i

Appendix: Data Model

You are given the data model below for a database on books, authors, editors and topics of books.

```
CREATE TABLE people (
    id                INT PRIMARY KEY
    , name            VARCHAR(100) NOT NULL
    , countryOfBirth  VARCHAR(100)
    , yearOfBirth     INT
    , numPublishers   INT ) ;

CREATE TABLE topics (
    id                INT PRIMARY KEY
    , title            VARCHAR(100) NOT NULL ) ;

CREATE TABLE books (
    id                INT PRIMARY KEY
    , title            VARCHAR(100) NOT NULL
    , publisher        VARCHAR(100) NOT NULL
    , isbn             NUMBER(13) NOT NULL
    , versionNo        INT NOT NULL
    , previousVersionId INT
        -- the id of the previous version of this book
    , publicationYear  INT
    , price            NUMBER(10,4)
    , copiesPublished  INT
    , FOREIGN KEY (previousVersionId) REFERENCES books (id)
) ;

CREATE TABLE bookAuthors (
    bookId            INT PRIMARY KEY
    , personId         INT PRIMARY KEY
    , PRIMARY KEY (bookId, personId)
    , FOREIGN KEY (bookId) REFERENCES books(id)
    , FOREIGN KEY (personId) REFERENCES people(id)
) ;

CREATE TABLE bookEditors (
    bookId            INT PRIMARY KEY
    , personId         INT PRIMARY KEY
    , PRIMARY KEY (bookId, personId)
    , FOREIGN KEY (bookId) REFERENCES books(id)
    , FOREIGN KEY (personId) REFERENCES people(id)
) ;

CREATE TABLE bookTopics (
    bookId            INT PRIMARY KEY
    , topicId          INT PRIMARY KEY
    , PRIMARY KEY (bookId, topicId)
    , FOREIGN KEY (bookId) REFERENCES books(id)
    , FOREIGN KEY (topicId) REFERENCES topics(id)
) ;
```