Database Systems, CSCI 4380-01 Exam #2 Thursday November 4, 2010 at 2 pm

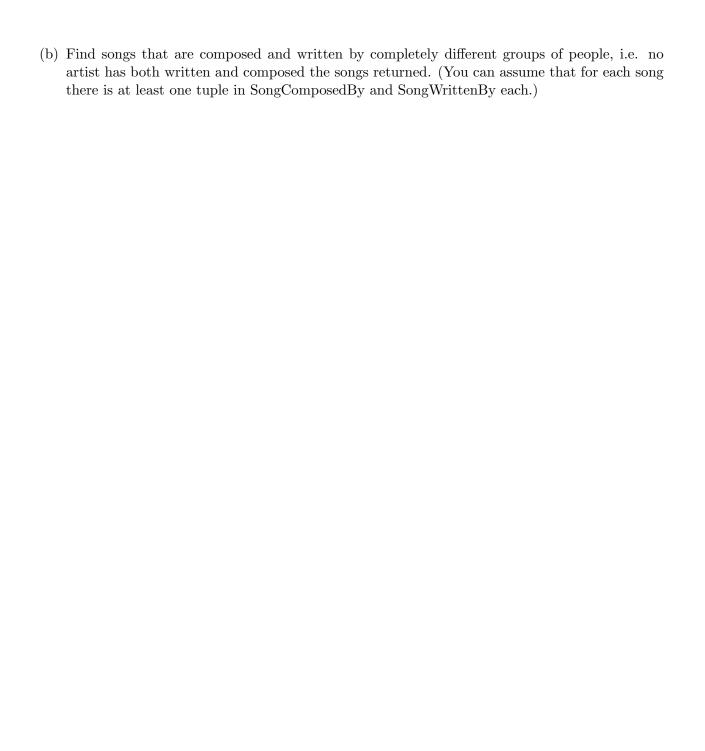
1a (12)	1b (12)	1c (12)	1d (12)	2 (14)	3 (16)	4 (14)	5 (8)	TOTAL

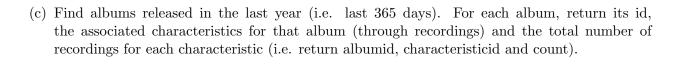
Note. The exam is open book and open notes. Use your own book and notes only, sharing is not allowed. Electronic gadgets are NOT allowed during the exam. Write your answers clearly, legibly and explain your reasoning as much as you can. If I cannot read or understand your answers, you will not get points.

In all SQL queries in this exam, unless stated otherwise, do not use views (i.e. using CREATE VIEW statements), triggers and other procedural elements. Make sure you use DISTINCT only when you have to.

Question 1 (12 points each). Write the following queries using SQL.

(a) Find all artists who have written or composed at least one of the songs they recorded. Return the id, name of the artist and the songs that they recorded and contributed to.





(d)	For each song, return its id, the total number of different artists that have recorded the song.	of times it has	been recorded a	and the number of

Question 2 (14 points). You are given the following bag relational algebra operation.

 $\begin{array}{rcl} R1 & := & (\Pi_{id \to songId} \; Songs) \times (\Pi_{id \to artistId} \; Artists) \\ R2 & := & SongComposedBy \; \cup \; SongWrittenBy \\ Result & := & \delta(\Pi_{artistid} \left((\delta \, R1) - (\delta \, R2) \right)) \end{array}$

- (a) Write what this query is computing in English (i.e. do not use words like select, project, join). (**Hint.** Set up a very small example database and show what it computes if you cannot express the query in English. This will help get you partial credit.)
- (b) Convert this bag relational algebra query to SQL.

Question 3 (16 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 (you can assume each operation executes only in the given tables, not consecutively). For each query, write a sentence explaining why the final contents of the tables are as given.

CRE	EATE TABLE ghi (
	id INT PRIMARY KEY
	, def_id INT NOT NULL
	, CONSTRAINT ghi_fk FOREIGN KEY def_id
	REFERENCES def(id)
	ON DELETE SET NULL ON UPDATE CASCADE) ;

abc					
id	a1				
1	'a'				
2	'b'				
3	'c'				

def						
id	abc_id					
1	1					
2	1					
3	2					

	ghi						
id	def_id						
1	1						
2	1						
3	2						
4	3						

(a) DELETE FROM abc WHERE id = 3;

	al	ос	def		ghi]
ĺ	id	a1	id	abc_id	id	def_id	

(b) DELETE FROM abc WHERE id = 1;

	al	ос		def		ghi
ic	l	a1	id	abc_id	id	def_id

(c) UPDATE abc SET id = 4 WHERE id = 2;

abc	abc		def	ghi		
id a1		id	abc_id	id	def_id	

(d) UPDATE def SET id = 4 WHERE id = 3;

al	abc			def	ghi		
id	a1		id	abc_id	id	def_id	

Question 4 (14 points). Create an E/R diagram for the following database.

Suppose you are creating a database for elections. For each election year, you would like to store the names of the candidates for each office and their party affiliation. The offices in consideration for this database are governor of a state, representative from a specific district of a state and senator of a specific state. Also store for each candidate the total votes they got, if they won and when the win was announced. Finally, for each office in question, store the name of the person holding that office and the start and end years of their term (for example Senator Obama was a senator for IL between 2005 and 2008). Of course, people can hold an office multiple times in different years.

Question 5 (8 points). Answer the following questions.

(a) What are the isolation levels that do not permit dirty reads? Explain why having dirty reads is a problem with one sentence.

(b) What does the following query compute?

```
CREATE FUNCTION checkX(inputId int) RETURNS boolean AS '
DECLARE
    currentNum int ;
    lastNum int ;
BEGIN
    currentNum = 0 ;
    lastNum = 0;
    FOR currentNum IN SELECT trackNum FROM Recordings
               WHERE albumId = inputId ORDER NY trackNum ASC LOOP
        IF currenNum <> lastNum +1 THEN
            RETURN FALSE;
        END IF ;
        lastNum = currentNum ;
    END LOOP ;
    RETURN TRUE ;
END ;
' LANGUAGE plpgsql;
SELECT
   DISTINCT R.albumId
FROM
   Recordings R
WHERE
   checkX(R.albumId) = false ;
```

Blank page for answers

Appendix

Suppose you are given the below data model for an application providing music services. This is the same data model that we have created in class and used in Exam #1 (except I did not include the data relating to the users).

Note that, albumId refers to Albums(id), songId refers to Songs(id), artistId refers Artists(id), recordingId refers to Recordings(id) and characteristingsId refers to Characteristings(id).

DATA MODEL.

Artists(<u>id</u>, name, bio, birthYear)

Albums(<u>id</u>, name, releaseDate)

Songs(id, name, lyrics)

Recordings(id, trackNum, length, songId, albumId)

Characteristics(<u>id</u>, name)

RecordingHasCharacteristics(recordingId, characteristicId)

AlbumsReleasedBy(albumId, artistId)

SongComposedBy(songId, artistId)

SongWrittenBy(songId, artistId)