Database Management Systems
Seminar 6

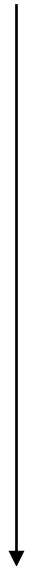
Model for the Practical Exam

I. Consider relation *Customer[CustomerID, FirstName, LastName, City, DateOfBirth]* and the interleaved execution below (in SQL Server).

There are no indexes on *Customer* and no other concurrent transactions. The value of *City* for the customer with *CustomerID 2* is *Timisoara* when T1 begins execution.

Choose the correct answer(s) for multiple choice questions 1 to 3.

T1	**T2**
BEGIN TRAN	
	BEGIN TRAN
SELECT City FROM Customer WHERE CustomerID = 2	
	UPDATE Customer SET City = 'Cluj-Napoca' WHERE CustomerID = 2
UPDATE Customer SET City = 'Bucuresti' WHERE CustomerID = 2	
	ROLLBACK TRAN
COMMIT TRAN	



1) T1 and T2 run under READ UNCOMMITTED. After the *COMMIT TRAN* statement in T1, the *City* value for the customer with *CustomerID* 2 is: (1p)
a. *Timisoara*
b. *Cluj-Napoca*
c. *Bucuresti*
d. NULL
e. none of the above answers is correct.

2) T1 runs under READ COMMITTED and T2 under REPEATABLE READ. After the *COMMIT TRAN* statement in T1, the *City* value for the customer with *CustomerID* 2 is: (1p)
a. *Timisoara*
b. *Cluj-Napoca*
c. *Bucuresti*
d. NULL
e. none of the above answers is correct.

3) T1 runs under REPEATABLE READ and T2 runs under READ COMMITTED. Then: (1p)
a. T1 doesn't acquire a shared lock for its SELECT statement.
b. The execution ends in a deadlock.
c. T2 needs an exclusive lock for its UPDATE statement.
d. T1 needs an exclusive lock for its UPDATE statement.
e. none of the above answers is correct.

II. Create a database for a MiniFacebook system. The entities of interest to the problem domain are: *Users*, *Pages*, *Likes*, *Categories*, *Posts*, and *Comments*. Each user has a name, current city and date of birth. A user can like multiple pages. The system stores the date of each like. A page has a name and a category, e.g., *sports*, *movies*, *music*, etc. A category also has a category description. Users write posts and comment on existing posts. A user's post has a date, text, and number of shares. A comment is anonymous, has a text, a date, and a flag indicating whether it's a top comment for the corresponding post.

a. Write an SQL script that creates the corresponding relational data model. (2p)

b. Create a Master/Detail Form that allows one to display the posts for a given user, to carry out <insert, update, delete> operations on the posts of a given user. The form should have a *DataGridView* named *dgvUsers* to display the users, a *DataGridView* named *dgvPosts* to display all the posts of the selected user, and a button for saving added / deleted / modified posts. You must use the following classes: *DataSet*, *SqlDataAdapter*, *BindingSource*. (2p)

c. Create a scenario that reproduces the non-repeatable read phenomenon on this database. Explain why the non-repeatable read occurs, and describe a solution to prevent this concurrency problem. Don't use stored procedures. (2p)

4) Submit a pdf file named *Group_LastName_FirstName.pdf* (e.g., *929_Ionescu_Ana.pdf*) that contains:

- the database diagram;

- the SQL script that creates the relational data model (*a*);

- the C# code that (*b*):

-- connects to the database;

-- fetches data into the application;

-- binds the DataGridViews such that whenever a different user is selected in *dgvUsers*, *dgvPosts* displays all his / her posts;

-- sends changes operated through *dgvPosts* back to the database;

- the SQL script that reproduces the non-repeatable read phenomenon (*c*).

Provide detailed comments explaining your code (both C# and SQL) and design choices.

(1p of)