

Practical Test

I. *Products* is a table in a SQL Server database with schema *Products*[*ProductID*, *Name*, *Value*]. The primary key is underlined.

ProductID is the search key of the clustered index on *Products*. The table doesn't have any other indexes.

Consider the interleaved execution below. There are no other concurrent transactions. The value of *Name* for the product with *ProductID* 3 is *Shirt* when T1 begins execution.

Answer questions 1-3 (each question has at least one correct answer)

T1	T2
BEGIN TRAN SELECT Name FROM Products WHERE ProductID = 3	
	BEGIN TRAN UPDATE Products SET Name = 'New Name' WHERE ProductID = 3
SELECT Name FROM Products WHERE ProductID = 3	
COMMIT TRAN	
	ROLLBACK TRAN

1. T1 runs under READ UNCOMMITTED and T2 under READ COMMITTED. After the execution of T1, the *Name* value shown by the SELECT queries is:

- Shampoo*
- New Name*
- '*New Name*'
- Shirt*
- None of the above answers is correct.

2. T1 and T2 run under REPEATABLE READ. After the execution of T1, the *Name* value shown by the SELECT queries is:

- Shampoo*
- New Name*
- '*New Name*'
- Shirt*
- None of the above answers is correct.

3. T2 runs under REPEATABLE READ. Then:

- T2 does not hold S locks.
- T2 avoids dirty reads and unrepeatable reads.
- T2 allows phantom reads.
- All of the above are correct.
- None of the above answers is correct.

II. Create a database for a system that manages a book shop. The entities of interest to the problem domain are: *Books*, *Promotions*, *Customers*. Books have a title, stock, a price per unit and type (drama, mystery, science, travel, etc.). Promotions are composed of one or more books and one book can be in a single promotion. Each promotion has a name, a start date and an end date. Customers have a name, phone number and an email address. Customers have two options for buying books: from the book shop or by ordering online.

1. Write an SQL script that creates the corresponding relational data model.
2. Create a Master/Detail Form that allows one to display the books for a given promotion, to carry out <insert, update, delete> operations on the books of a given promotion. The form should have a *DataGridView* named *dgvPromotion* to display the promotions, a *DataGridView* named *dgvBooks* to display all the books of the selected promotion, and a button for saving added / deleted / modified books. You must use the following classes: *DataSet*, *SqlDataAdapter*, *BindingSource*.
3. Create a scenario that reproduces the non-repetable read concurrency issue on this database. Explain why the non-repeatable read occurs, and describe a solution to prevent this concurrency issue. Don't use stored procedures.

I. 1	1p
2	1p
3	1p
II. 1	2p
2	2p
3	2p
	1p of