Assignment 8 & 9 – Transactions

Dates of work: `Oct 11, 2022`, `Oct 18, 2022`;

Submission of complete assignment (code + report of analyses): `Oct 30, 2022`;

Viva: `Nov 1, 2022` & `Nov 2, 2022`; Total Points: 180

Questions:

1. Use the multi-table database for a library from the previous assignment.
   a. Simulate an environment that allows multithreaded database access for multi-transaction handling (References provided below) **(10 points)**
   b. Simulate a deadlock situation derived from multiple transactions attempting to access & update the same tuple - **(10 + 15 * 3 = 55 points)**
      i. Justify that the situation is indeed one of a deadlock
      ii. If you had to choose a victim to resolve the deadlock, what would your algorithm be? [Note: You are allowed to think of a mix of priority mechanisms + lock + timestamp mechanisms for deadlock resolution]
      iii. Attempt execution of your algorithm for deadlock avoidance
      iv. Comment on the conflict serializability of your algorithm

2. Simulate a deadlock situation derived from multiple transactions attempting to access & update the same table - **(10 + 15 * 3 = 55 points)**
   a. Justify that the situation is indeed one of a deadlock
   b. If you had to choose a victim to resolve the deadlock, what would your algorithm be?
   [Note: You are allowed to think of a mix of priority mechanisms + lock + timestamp mechanisms for deadlock resolution]
   c. Attempt execution of your algorithm for deadlock avoidance
   d. Comment on the conflict serializability of your algorithm

3. Create a real-life scenario in which an in-house application returns an error to the users, and users notify the development team about this error. The development team realizes that it is a deadlock issue, but they could not find the main reason for the problem. Under these circumstances, the team decides to receive consultancy service from an experienced database administrator. Do simulation for the same. **(10 Points)**

4. Take the following scenario & do the simulation for the same. Suppose we have two database sessions called A and B. Let's say that session A requests and has a lock on some data – and let's call the data Y. And then session B has a lock on some data that we will call Z. But now, let's say that session A needs a lock on data Z in order to run another SQL statement, but that lock is currently held by session B. And, let's say that session B

needs a lock on data Y, but that lock is currently held by session A. This means that session B is waiting on session A's lock and session B is waiting for session A's lock. (10 Points)

5. Simulate a deadlock situation to prove that the deadlock state can be changed back to stable state by using Rollback statement. **(10 Points)**

6. Simulate the Wait-die situation in deadlock for: When transaction Ti requests a data item currently held by Tj, Ti is allowed to wait only if it has a timestamp smaller than that of Tj (that is, Ti is older than Tj). Otherwise, Ti is rolled back (dies). **(10 Points)**

7. Simulate a sequence of transactions that leads to an irrecoverable state. Comment on how the transactions could be modified to make it recoverable and simulate the recoverable sequence. **(10 Points)**

8. Simulate a sequence of transactions that requires a cascading rollback when the latest transaction fails. Convert the sequence to a cascadeless schedule and simulate the cascadeless schedule. **(10 Points)**

References:
1. https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15
2. https://www.ibm.com/docs/en/db2/11.1?topic=designing-concurrent-transactions-multi-threaded-database-access
3. https://www.columbia.edu/sec/acis/db2/db2a0/db2a082.htm
4. https://www.sqlite.org/threadsafe.html

Assessment Rubric for Submitted Work - Evaluation per answer (Viva & Analyses):

Proper understanding of question and work done accordingly:	80 - 95%
Attempt to work beyond what has been asked, with in-depth understanding: (Definite contender for full score for question)	95 - 100%
Vague understanding, bursts of in-depth answers:	70 - 80%
Vague understanding, bursts of broad conceptual answers:	50 - 70%
Weird hash of work submitted, some understanding:	40 - 50%
No understanding, just work submitted: (Probable plagiarism)	0

Submission Rubric:

Within 2 days of Deadline:	No penalty
Within 5 days of Deadline:	30% penalty
Within 7 days of Deadline:	50% penalty
After 7 days of Deadline:	Will not be evaluated

———————————————