PHASE 1: Transaction management and concurrency control

* 1. Define the term: Transaction
  2. Name and define the transaction properties
  3. Discuss what Concurrency Control is and the objective thereof
  4. Explain the difference between pessimistic locking and optimistic concurrency control. Provide one situation where each would be appropriate.
  5. Consider two transactions running concurrently:   
     **T1: UPDATE Account SET Balance = Balance - 500 WHERE Acc\_No = 123; T2: UPDATE Account SET Balance = Balance + 500 WHERE Acc\_No = 123;**   
     Describe the potential concurrency problem. Explain how the two-phase locking protocol (2PL) can prevent this problem.
  6. Discuss what a deadlock is.
  7. Explain the three main problems associated with concurrency control
  8. Name the FIVE levels at which locking can take place in a DBMS (lock granularity).
  9. Explain the three main problems associated with concurrency control.
  10. Distinguish between a shared lock and an exclusive lock, and under what circumstances would these two locks be granted
  11. Name and briefly describe FOUR properties of transactions.
  12. Explain what the concurrency control with Time Stamping method entails, also refer to the two properties of the Time Stamping method
  13. Explain what Two-Phase Locking entails and briefly describe the two phases.
  14. **Explain what a transaction log is and describe what its function is**

PHASE 2:

1. The DBMS processes queries in three phases. Name those phases, and describe what is accomplished in each phase
2. Most query optimization techniques are designed to make the optimizer’s work easier. What factors should you keep in mind if you intend to write efficient conditional expressions in SQL code?
3. Explain why indexes are used for query optimization
4. Discuss the difference between SQL performance tuning and DBMS performance tuning.
5. Describe the objective of a query optimisation routine in a DBMS and list the costs associated with the execution of a request that it aims to minimise
6. .The DBMS query optimiser chooses an execution plan for the following query:   
   SELECT C.CustomerName, O.OrderDate   
   FROM Customer C, Orders O   
   WHERE C.CustID = O.CustID AND O.OrderDate > '2025-09-16';  
     
   a) Suggest one way the optimiser could rearrange the query to improve performance.  
   b) Why might using an index on OrderDate improve query performance?